The Report committee for Nikhil Jayant Inamdar

Certifies that this is the approved version of the following report:

Educational Shaking Table Modules for

Earthquake Engineering

APPROVED BY

SUPERVISING COMMITTEE:

Supervisor: ______________________________

Wassim Ghannoum

____

Lance Manuel

Educational Shaking Table Modules for Earthquake Engineering

by

Nikhil Jayant Inamdar

Report

Presented to the Faculty of the Graduate School

of the University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

December 2010

Dedication

I dedicate this report to my mom, Mrs. Nirmala Inamdar and my dad, Mr. Jayant Inamdar, for their extensive support, patience, love and understanding.

Educational Shaking Table Modules for Earthquake Engineering

by

Nikhil Jayant Inamdar

The University of Texas at Austin, 2010

SUPERVISOR: Wassim M. Ghannoum

The goal of the project is to develop, build, and test a modular steel structure that can be tested on an educational shaking table to demonstrate structural dynamic and earthquake engineering principles. The advantage of the structure is that it can be tested into its non-linear range and yielded parts can easily be replaced for subsequent tests. The steel modular structure represents a multi-story moment resisting frame and is comprised of sheet metal beams and columns bolted to "rigid" steel angles. This structure is tested on a unidirectional shaking table, viz. "Quanser Shake Table II". The structure is designed to achieve a specific mode of failure through non-linear analysis. A non-linear pushover analysis is carried out to determine stiffness and strength of the structure as well as potential hinge locations. Eigen-value analysis is undertaken to determine all the natural periods and frequencies that will help in understanding its dynamic response. The structure is analyzed and tested for various ground motions to study the effects of an earthquake on a multi-storied frame. Educational modules provide a set of experiments that can be easily performed on the test structure.

Table of Contents

List of Figures

List of Tables

1. Quanser Shake Table II

1.1 Introduction

The Shake Table II[1] developed by Quanser is a unidirectional shaking table with a stage operated by a powerful brushless motor. The table can achieve a maximum acceleration of 2.5g when the stage is not loaded. The stage is capable of moving 3 inches in each direction; a total linear stroke of 6 inches. Table maximum payload is 15kg or 33lb. The entire system is comprised of a Universal Power Module (UPM), a data-acquisition (DAC) card, a PC running the QuaRC control software, and the Shake Table II itself; as shown in the Figure 1. This first section is meant to familiarize the reader with table usage. The details regarding shaking table components, design, and setup discussed below follow closely the Shake Table II manual provided by Quanser[2].

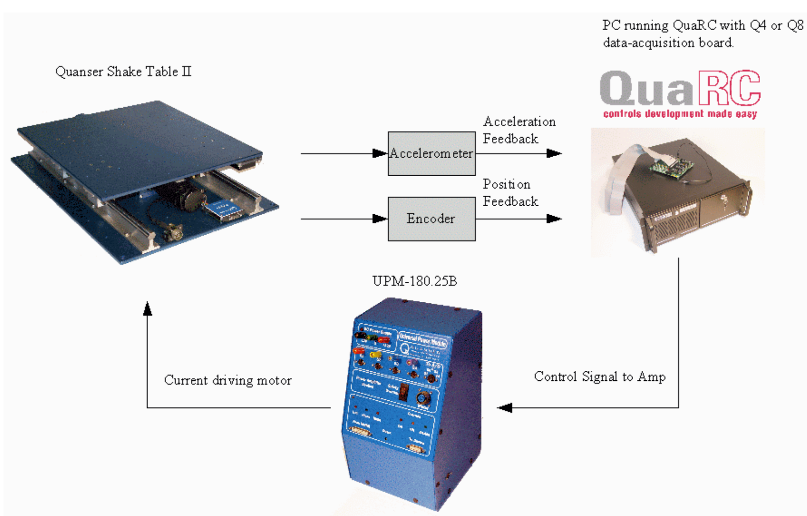


Figure 1: Overview of major system components[2]

1.2 **Shake Table II Specifications**

Parameter Description	**SI Value**	**Units**	**IM Value**	**Units**
Motor armature resistance	2.94	ohm		
Motor current-torque constant	0.360	Nm/A	3.2	lb.in/A
Motor back-emf constant	0.2034	V/(rad/s)	23.4	V/k_{rpm}
Ball-screw pitch	0.0127	m/rev	0.5	in/rev
Preload mass	7.74	kg	17.1	lb
Maximum payload mass	15.0	kg	33.0	lb
Mass of shaking table system	27.2	kg	60.0	lb
Dimension of top stage Dimension of bottom stage	0.46x0.46 0.61x0.46	m^2	18.0x18.0 24.0x18.0	in^2
Height from bottom to top stage	12.4	cm	4.875	in
Maximum stroke position	76.2	mm	3.0	in
Maximum linear velocity of stage	664.9	mm/s	26.18	in/s
Maximum linear force of stage	708.7	N	159.3	lb
Maximum linear acceleration for 0 kg load	2.5	g		
Encoder sensitivity gain	3.1006	μm/count	1.22E-04	in/count
Accelerometer sensitivity gain	-1	g/V		
Dynamic load capacity of ball nut	12000	N	2697.6	lb
Life expectancy of ball nut at full load	6.35E+08	m	2.50E+10	in
Life expectancy of linear bearing	6.35E+06	m	2.50E+08	in
Load carrying capacity of linear bearings	131.5	kg	290	lb
Operational Bandwidth	20	Hz		

Table 1: Shake Table II Specifications[2]

1.3 System Setup

The following hardware and software are required-

Power Amplifier: Quanser UPM 180-25B

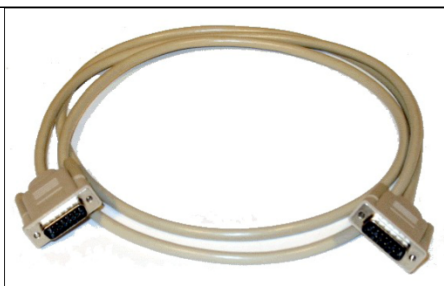
Data Acquisition Board: Quanser Q4 board

Shaking Table: Quanser Shake Table II

Real-Time Control Software: QuaRC 2.1[2]

1.4 Cable Nomenclature

The various connector cables that are provided with the Shake Table II system are listed in Table 2.

Cable	Description
	The "Motor" cable corresponds to the 3-phase motor power leads. This cable is designed to connect from the Quanser's Universal Power Module model 180-25B (i.e. the output of the power module after signal amplification) to the brushless DC motor of the shaking table.
	The "To Device" cable is a DB15 cable that connects the shaking table circuit board to the UPM. It carries to the UPM the three limit sensors' signals and the motor encoder signals. It also supplies the DC power required by the different sensors.

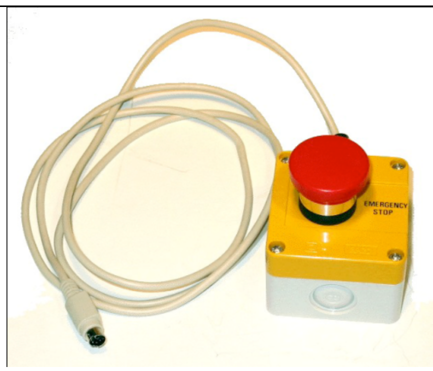
	The "From MultiQ" 25-pin parallel cable connects the UPM to the data acquisition card terminal board. It is compatible with Quanser's quick-connect system. It carries the motor encoder signals, limit sensor signals, calibrate signal, and the S1, S2, S3, and S4 analog signals from the UPM. From the DAC, the cable carries the control signal (to be amplified and sent to the motor) as well as they calibrate and enable digital signals.
	The "Emergency Stop" cable has a 6-pin-mini-DIN connector that connects to the side of the UPM. The UPM is enabled when the safety pushbutton switch is not pressed.
	The "From Analog Sensors" cable is a 6-pin-mini-DIN-to-6-pin-mini-DIN cable that can be used to connect any potential plant sensor to the UPM such as accelerometers. It can provide a ±12VDC bias to analog sensors and carry their voltage signals to the DAC terminal board via the UPM.

Table 2: Cable Nomenclature[2]

1.5 Detailed Wiring Procedure

This section describes in detail how to connect the Shake Table II, the power amplifier (UPM-180-25B) and the Q8 Extended Terminal Board (this is the external board that is attached to the PC internal Q4 data acquisition card). The connections are identified in Figures 2 with a corresponding identification number. Follow these steps to wire the ST II system:

1. **Connect cable #1** from the "Table X" connector on the Quanser Q8 Extended Terminal Board, shown in Figure 2, to the "From MultiQ" connector on the Quanser UPM-180-25B, as depicted Figure 2. This connection is done using the parallel cable shown in Table 2.
2. **Connect cable #2** from the "To Device" connector located on the UPM front panel, as shown in Figure 2, to the ST II circuit board illustrated in Figure 2. This cable used in this connection is the DB15 cable described in Table 2 and pictured in Figure 2. It carries all three proximity sensor signals, the motor encoder signals, and the brushless motor hall sensor signals to the UPM-180-25B unit. The motor encoder is used to calculate the linear position of the cart and is denoted by the variable *x*.
3. **Connect cable #3** into the side of the UPM, as presented in Figure 2. Cable #3 is the "Emergency Stop" cable described in Table 2 and illustrated in Figure 2. The UPM is active if and only if the remote E-Stop switch is *depressed*.

4. **Connect cable #4** from the "Motor" connector located on the UPM front panel, as depicted in Figure 2, to the "Motor" connector on the shaking table, as shown Figure 2. The motor leads connector is component #13 in Figure 2. This connection is done using the "Motor" cable described in Table 2 and illustrated in Figure 2. It carries the required 3-phase power to the brushless motor.
5. **Connect cable #5** from the analog connector on the accelerometer mounted on the shaking table, see Figure 2, to the “S1” connector on the front panel of the UPM-180-25B. **Ensure the UPM is not powered when making this connection.** It carries the acceleration measured by the accelerometer.
6. A summary of the wiring procedure is given in table 3.

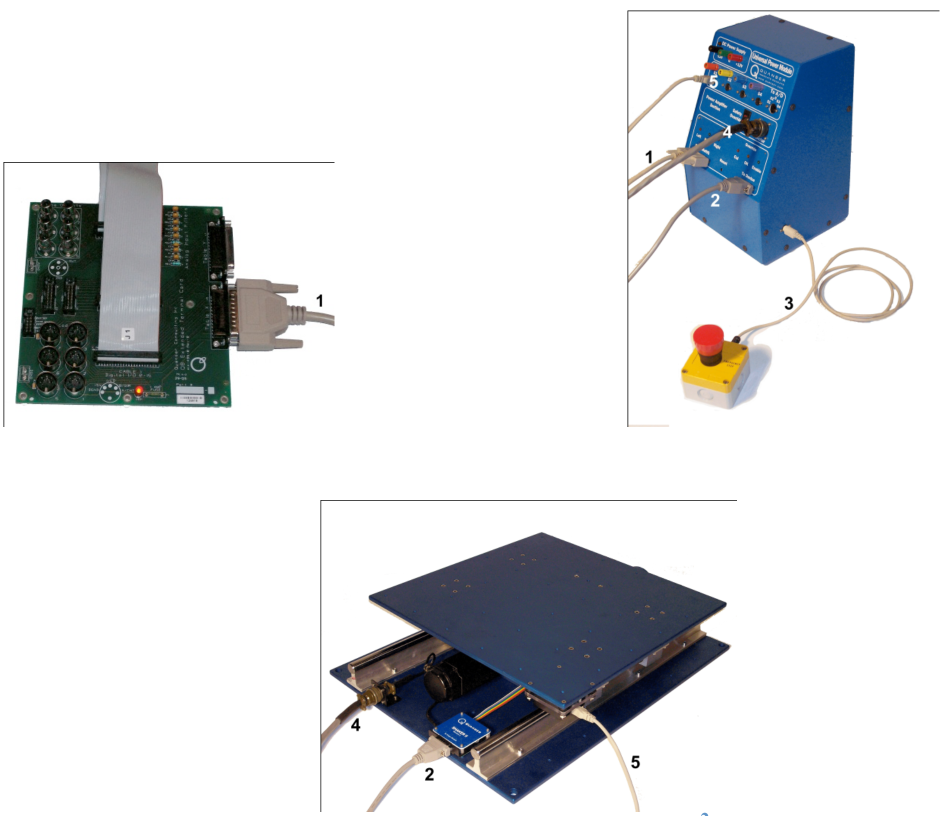


Figure 2: Cable Connections[2]

1.6 Wiring Summary

Cable #	Cable Type	From	To	Function
1	25-pin Parallel Cable	"Table X" on the Q8 Terminal Board	"From MultiQ" on the blue UPM-180-25B	Drives the amplifier to move the stage and receives the accelerometer, stage encoder, calibration, and limit detector signals from UPM.
2	15-pin Parallel Cable	"To Device" on the blue UPM-180-25B	Circuit board on the Shake Table II	Receives the encoder and limit detector signals from the shaking table.
3	"Emergency Stop" Cable	E-Stop Switch	UPM E-Stop Connector	Carries the emergency stop signal.
4	4-pin Motor Cable	"Motor" Connector on the UPM-180-25B	Motor connector on the Shake Table II	Connects the shaking table motor leads to the amplifier on the UPM.
5	Analog Cable: 6-pin-mini- DIN to 6-pin mini-DIN	"S1" Connector on the on the UPM-180-25B	Accelerometer on the Shake Table II	Carries the acceleration signal of the stage to the UPM.

Table 3: Shake Table II Wiring Summary[2]

1.7 Running the Shaking Table Software

Quanser provides two software options to run the shaking table.

1.7.1 Shake Table II Software - Simple User-Friendly Software

The first is a simple software with limited functionality that presents a user-friendly graphical user interface (GUI), through which, the table can directly be operated. The user can run this GUI by running the icon named "Shake Table II Software.exe" on the desktop. When the software is run, it automatically initializes the UPM and calibrates the table to home position (position at center of stroke). Through the software GUI, a user can run a sine wave, a sweep wave (chirp), or two pre-loaded ground motions from the 1994 Northridge and 1995 Kobe earthquakes. These ground motions are the 90 direction motion recorded at the Sylmar-Olive View station during the Northridge earthquake and the 00 direction motion recorded at the HIK station during the Kobe earthquake. The amplitude and frequency of the wave inputs can be varied using the controls available in the GUI. The ground motion records cannot be altered or scaled in this simplified software.

The Shake Table II Software has the GUI interface shown in Figure 3.

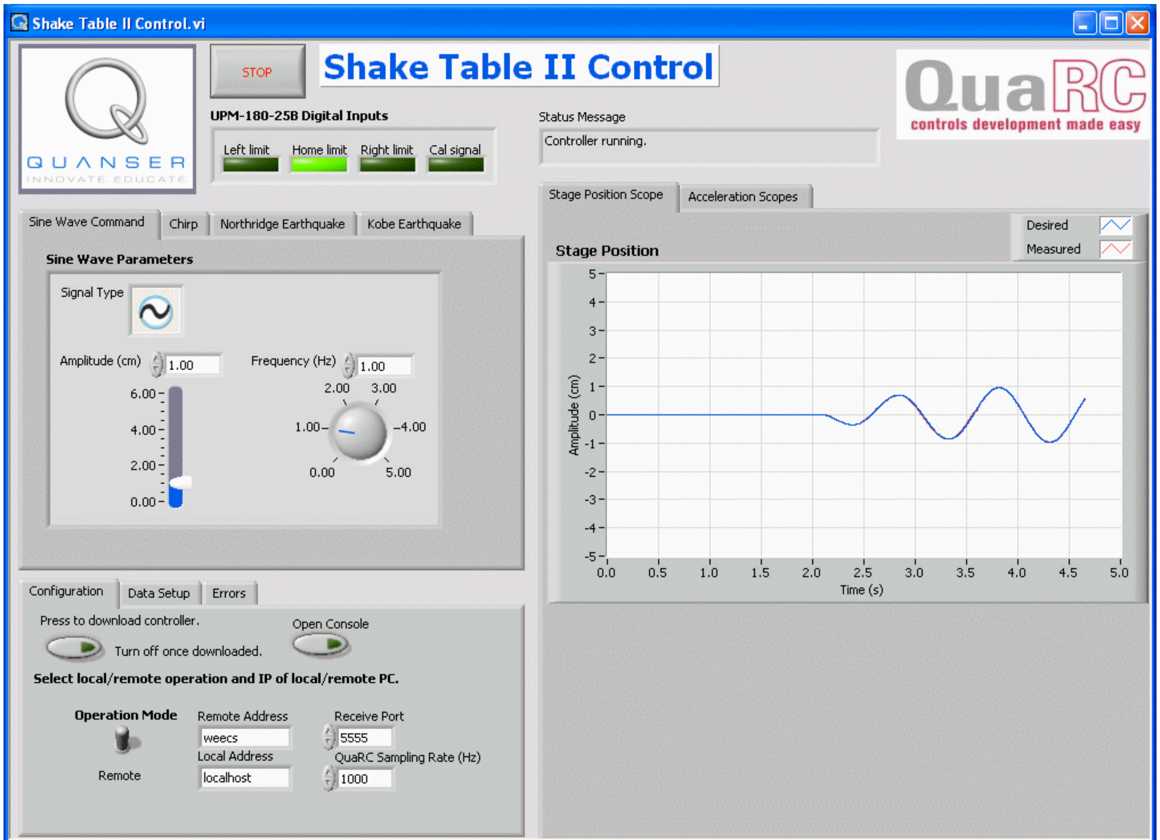


Figure 3: Shake Table II Software[2]

1.7.2 QuaRC Software and Controllers

The range of frequencies and ground motion records are limited in the simplified software. The software QuaRC that runs in conjunction with MATLAB-Simulink[3] provides the user with full control of the table. These controllers can direct the shaking table to move to any ground motion or wave function that is within its physical limits.

There are five standard controllers (methods of control) for the Shake Table II:

1. Initializing the UPM
2. Calibrating the table, i.e. moving stage to the home position
3. Running a sine wave

4. Running a sine sweep
5. Running a predefined trajectory such as an earthquake or sine wave

The following lists the Simulink models used to generate QuaRC controllers along with a short description.

Name	**Description**
q_boot_upm.mdl	Initializes the UPM-180-25B to make the amplifier ready-to-be-enabled. This has to be done prior to performing any of the ST II experiments.
q_cal.mdl	Returns the stage to the home position. The stage should be at the home position before running any of the experiments.
q_sine.mdl	Position of the stage tracks a sine wave with an amplitude and frequency set by the user.
q_sweep.mdl	Sends a sine sweep, i.e. chirp signal, to the shaking table for generating the frequency response.
q_data.mdl	Sends a predefined sine wave or an earthquake, e.g. Kobe or Northridge.

Table 4: Simulink models used to run on the ST II system[2]

To run these controllers, QuaRC controllers folder has to be installed on the PC hardrive and the MATLAB-Simulink software has to be loaded. The working directory in

MATLAB has to be set to the folder with Shake Table controllers such that the current directory looks as the one shown in Figure 4.

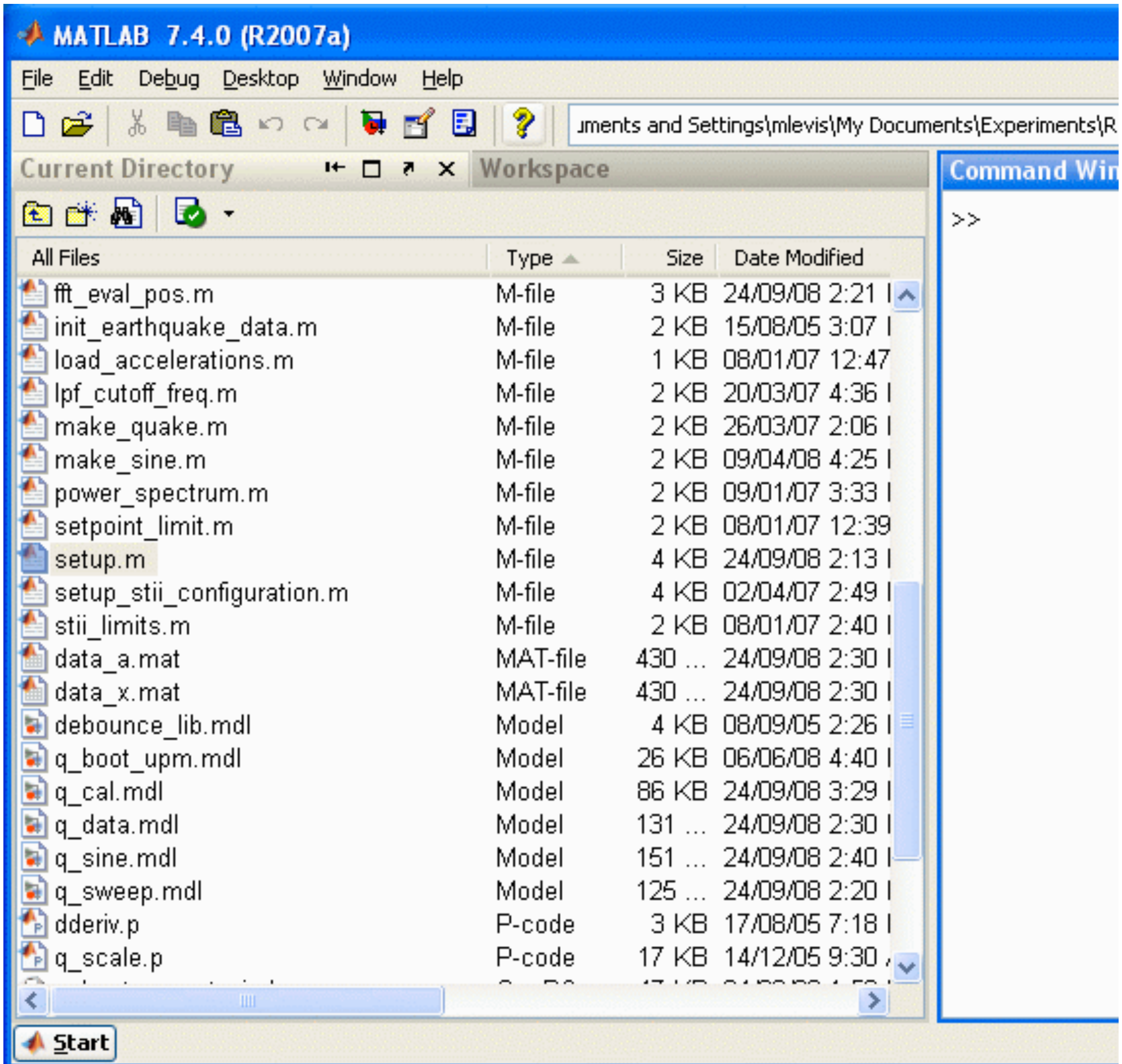


Figure 4: View of Current Directory window in MATLAB[2]

Follow this procedure to run the shaking table using QuaRC Controllers:

1. Double-click on setup.m MATLAB script. The window shown in figure 5 should appear:

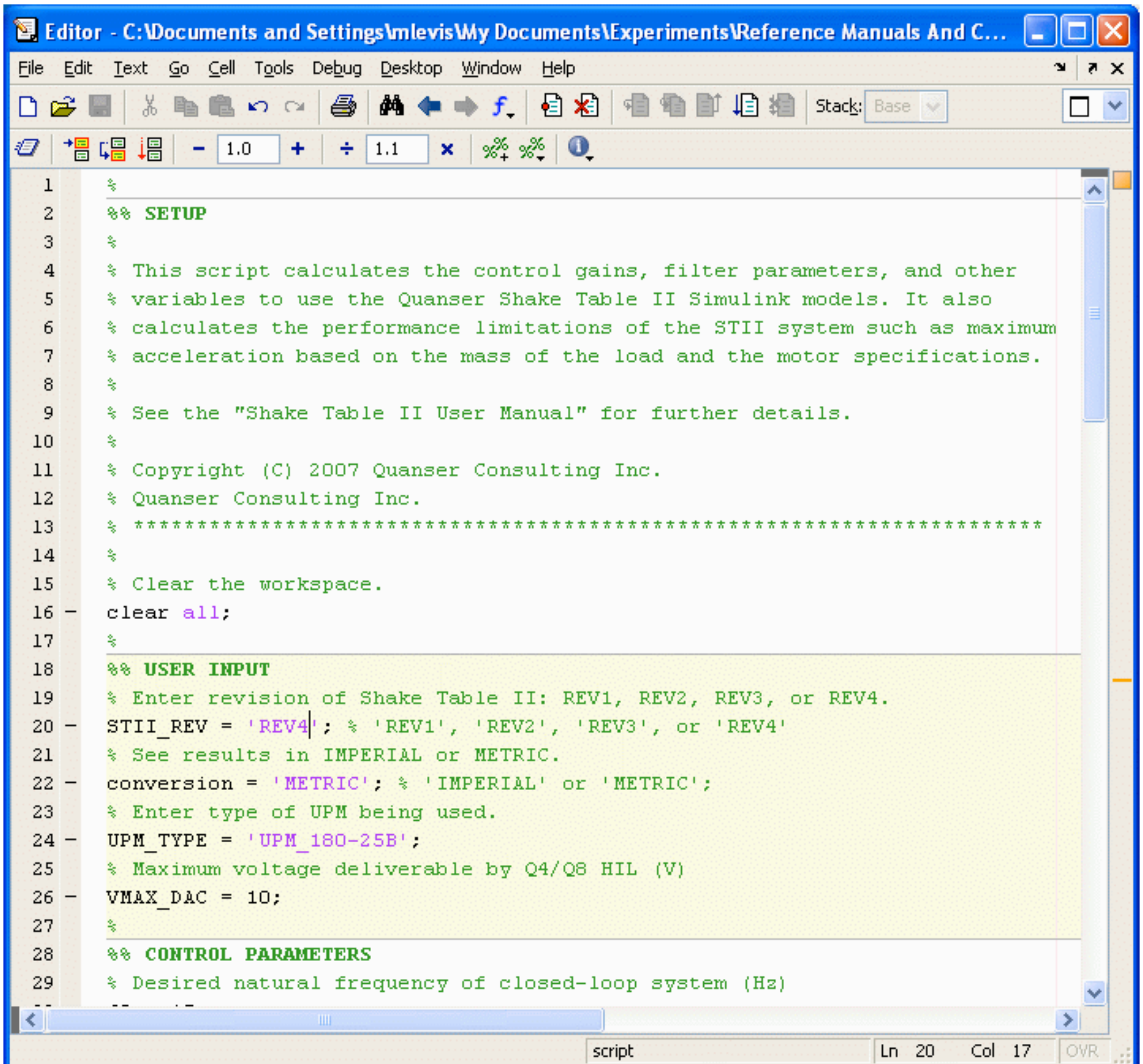


Figure 5: Shake Table II MATLAB setup.m script[2]

The user can modify the units in the input by setting it to either "METRIC" or "IMPERIAL" as shown in the file. It is advised not to modify any other settings.

Run the setup.m MATLAB file, this loads all the required parameters in MATLAB workspace. Note that before running any QuaRC Controller, setup.m file should be run only once. These controllers can be accessed from a folder named “QuaRC Controllers” installed on the desktop.

2. Initializing the UPM:

When the blue power amplifier UPM-180-25B is first powered, its Left and Right LEDs should be blinking, which means UPM is not yet ready for use. Ensure that the safety override switch is in off position and the red Stop button is connected. To initialize the UPM, q_boot.mdl is run (from MATLAB), after which the two Right and Left LEDs stop blinking and the message shown in Figure 6 appears.

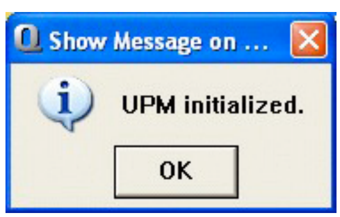


Figure 6: UPM initialization Message[2]

Note: DAQ Configuration has to be done if you are using a Q4 acquisition board. It can be done using the HIL Initialize block present in each of the Simulink Diagrams. By default, all the Simulink diagrams supplied are setup for the Q8 Hardware-in-the-loop (HIL) board. As the Q4 board accompanies the University of Texas shaking table, the user should double-click on the HIL Initialize block shown in Figure 7 and change the board type to Q4 as explained in Figure 8. Save the Simulink diagram to commit the board type changes. Once the board type is changed and saved, there is no need to change it again in the future.

Figure 7: HIL Initialize Block

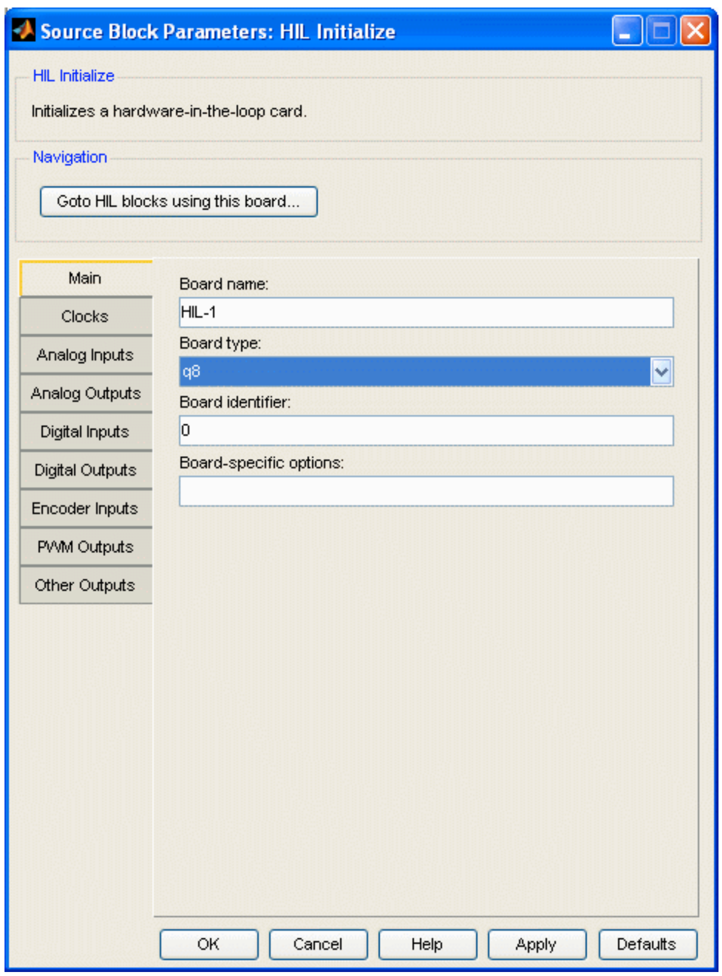


Figure 8: Setting the Board Type in the HIL Initialize Block[2]

Next run the q_boot.mdl Simulink diagram to generate the Controller needed by QuaRC. After running the q_boot.mdl Simulink diagram, run the QuaRC program and click the "Build from the Simulink diagram" menu bar. To run the QuaRC Controller, either click on "Run from the menu bar" or select the "*Connect To Target* and *Run"* buttons from the tool bar. This will run the q_boot.mdl QuaRC Controller to initialize the UPM.

3. Calibrating the Stage:

Before running any experiment, the shaking table should be centered at the home position. Home LED is lit after the calibration process is successful. After, initializing the UPM, from MATLAB run q_cal.mdl Simulink diagram to run the calibration QuaRC Controller. When the controller is run, the *Cal*, *OK*, and *Enable* LEDs on the front panel of the UPM should all be lit and the stage should begin moving. If the *Left* or *Right* limit sensor was already triggered, then the stage begins to immediately move towards the center. If no limit switch was initially triggered, then the stage will begins to move towards *Left* limit sensor. Once the *Left* limit sensor is hit, the stage reverses its direction and begins moving towards the mid-stroke position. The stage stops moving when the *Home* limit switch is triggered (the *Home* LED on the UPM will go ON). When complete, the message in Figure 9 will be shown.

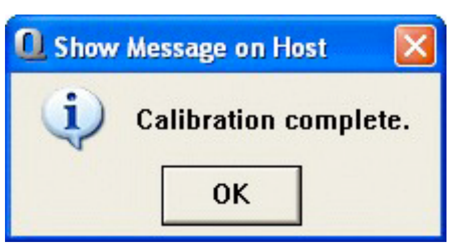


Figure 9: Calibration complete Message[2]

4. Running an Experiment:

After calibration is complete, the user can run experiments such as sine wave, sweep wave or a ground motion on the shaking table. For all the experiments, Quanser provides specific MATLAB scripts and corresponding Simulink Diagrams. How to run an earthquake ground motion will be explained in detail.

Follow this procedure to run a sample earthquake:

i. After initialization of the UPM and calibration of the table, load the q_data.mdl Simulink Diagram by double-clicking the file on the MATLAB working directory. The diagram shown in Figure 10 will appear, but do not attempt to build it.
ii. Before building the QuaRC controller, the command position and acceleration must first be loaded into the MATLAB environment. For running a ground motion, open and run the make_script.m MATLAB script.
iii. Now build and run the q_data.mdl.

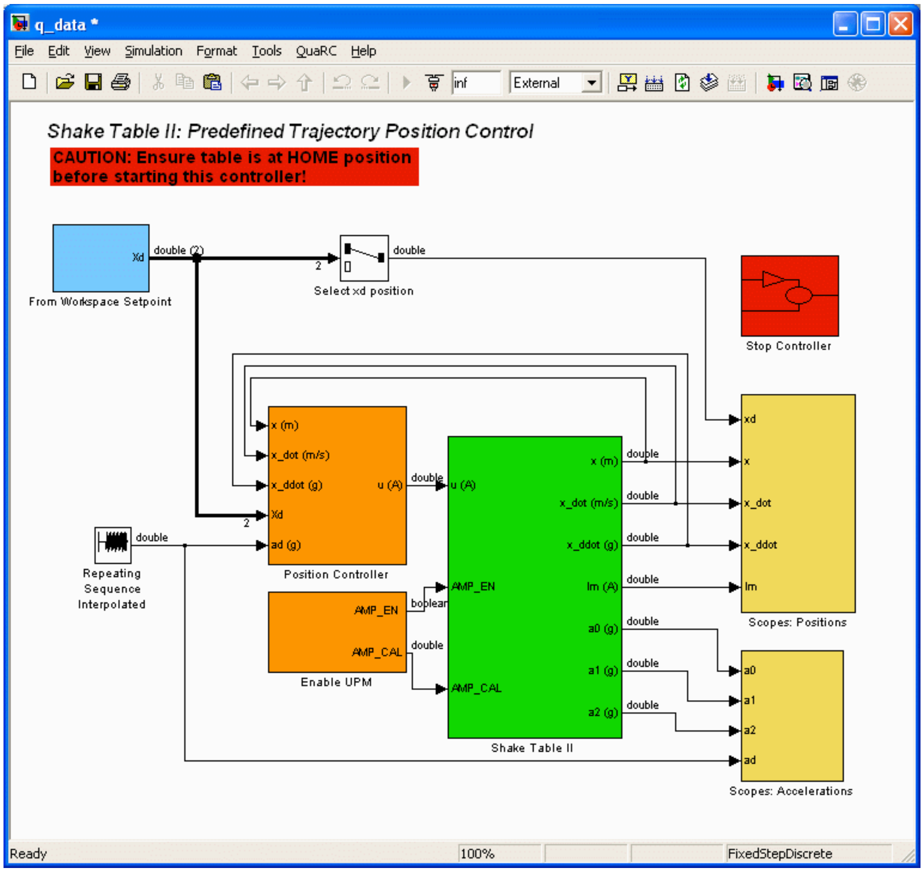


Figure 10: Simulink diagram used with QuaRC to run pre-defined trajectories, e.g. earthquakes[2]

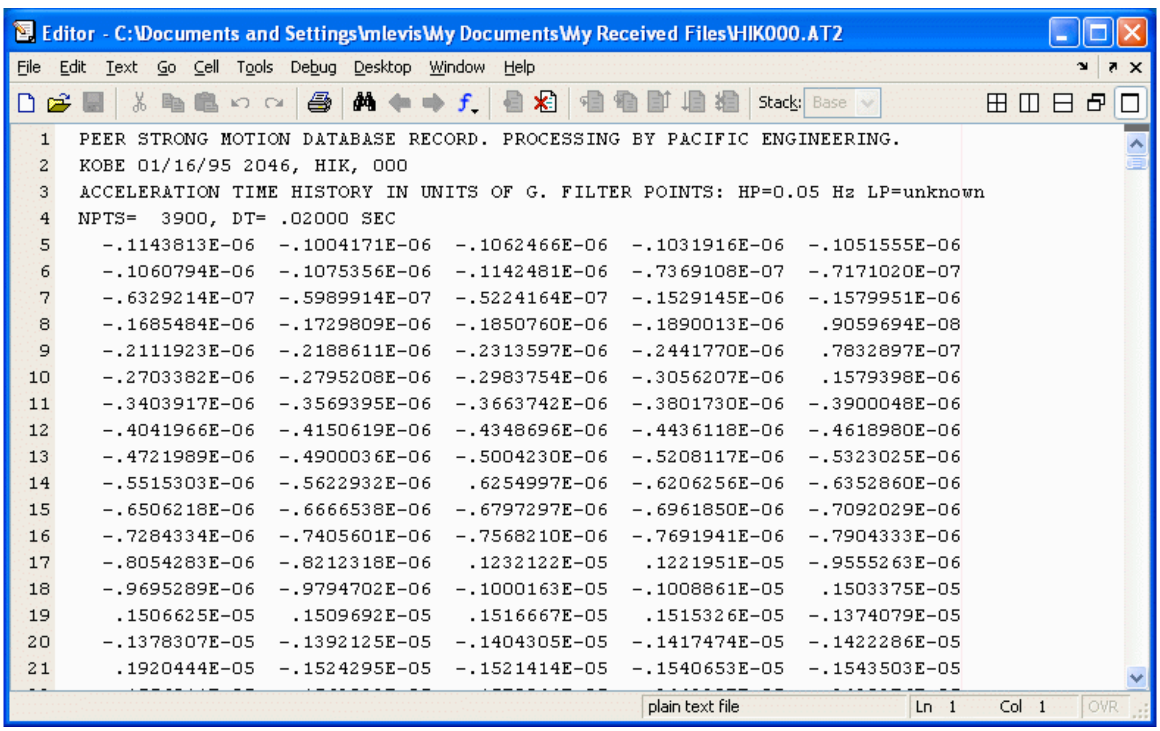
The make_quake.m MATLAB script, allows a user to assign earthquake data of any ground motion to the shaking table system.

1.7.3 Finding a Ground Motion Record

1. There are a variety of resources on the Internet where ground motion data can be downloaded. Two example sources are the Pacific Earthquake Engineering Research Center (PEER) Strong Motion Database website at http://peer.berkeley.edu/smcat/search.html from the University of California and

the Lamont-Doherty Earth Observatory of Columbia University at http://www.ldeo.columbia.edu/nceer/ strongmo.html. On the PEER website, each earthquake has various measurement stations and each station contains displacement, velocity, and acceleration data of the tremor for different directions. IF a PEER ground motion record is downloaded, save the AT2 text file in the *STII\Lab Files\mdl* folder on the shaking table PC. When opened in MATLAB Editor, the ground motion file should appears as shown in Figure 11.

2. Once the AT2 text file is saved, write the file name in the make_quake.m MATLAB script in order to run the ground motion on the table.




```
PEER STRONG MOTION DATABASE RECORD. PROCESSING BY PACIFIC ENGINEERING.
KOBE 01/16/95 2046, HIK, 000
ACCELERATION TIME HISTORY IN UNITS OF G. FILTER POINTS: HP=0.05 Hz LP=unknown
NPTS=  3900, DT= .02000 SEC
  -.1143813E-06  -.1004171E-06  -.1062466E-06  -.1031916E-06  -.1051555E-06
  -.1060794E-06  -.1075356E-06  -.1142481E-06  -.7369108E-07  -.7171020E-07
  -.6329214E-07  -.5989914E-07  -.5224164E-07  -.1529145E-06  -.1579951E-06
  -.1685484E-06  -.1729809E-06  -.1850760E-06  -.1890013E-06   .9059694E-08
  -.2111923E-06  -.2188611E-06  -.2313597E-06  -.2441770E-06   .7832897E-07
  -.2703382E-06  -.2795208E-06  -.2983754E-06  -.3056207E-06   .1579398E-06
  -.3403917E-06  -.3569395E-06  -.3663742E-06  -.3801730E-06  -.3900048E-06
  -.4041966E-06  -.4150619E-06  -.4348696E-06  -.4436118E-06  -.4618980E-06
  -.4721989E-06  -.4900036E-06  -.5004230E-06  -.5208117E-06  -.5323025E-06
  -.5515303E-06  -.5622932E-06   .6254997E-06  -.6206256E-06  -.6352860E-06
  -.6506218E-06  -.6666538E-06  -.6797297E-06  -.6961850E-06  -.7092029E-06
  -.7284334E-06  -.7405601E-06  -.7568210E-06  -.7691941E-06  -.7904333E-06
  -.8054283E-06  -.8212318E-06   .1232122E-05   .1221951E-05  -.9555263E-06
  -.9695289E-06  -.9794702E-06  -.1000163E-05  -.1008861E-05   .1503375E-05
   .1506625E-05   .1509692E-05   .1516667E-05   .1515326E-05  -.1374079E-05
  -.1378307E-05  -.1392125E-05  -.1404305E-05  -.1417474E-05  -.1422286E-05
   .1920444E-05  -.1524295E-05  -.1521414E-05  -.1540653E-05  -.1543503E-05
```

Figure 11: Raw earthquake data file HIK000.AT2 shown when opened in MATLAB Editor[2]

2. Modular Structures

In this chapter, the design, analysis, and testing of the modular structure will be discussed.

2.1 Design

When designing the modular structure, shaking table limitations were of primary concern. The following are the main ones:

Maximum Payload: **15 kg**

Operational Bandwidth: **20 Hz**

Maximum Linear Acceleration: **2.5g**

The structure was designed in such a way that at least three to four natural modes can be excited using the shaking table. This approach is akin to performance-based design. Several structures were considered prior to settling on the one that was built. Section dimensions were selected to obtain a predetermined response from the structure. The following three cases were considered:

1. Low rise structure – Weak Column, Strong Beam
2. Low rise structure – Strong Column, Weak Beam
3. High rise structure - Strong Column, Weak Beam

The material selected was ASTM A36 / A36M - 08 carbon structural steel, with a specified yield stress of 36 ksi.

2.2 Sizing of Sections

The sections used for the modular structure are in the form of thin steel plates (or sheet metal). Plate sections were selected considering weight limitations and for their flexibility; which accentuates deformations for better visualization. Spans, floor heights, and plate thicknesses were selected considering the structure's weight, natural periods, and buckling potential. The prime motive was to achieve a design where at least the first three modes of vibration could be excited by the shaking table. It was critical not to have buckling or excessive second order effects in the structure. The ratio of buckling load to axial load of the structure (buckling ratio) was maintained above 2. Preliminary elastic analyses were conducted to finalize the geometry of the structure following which non-linear analyses were conducted to better characterize the behavior of the structure. Only the non-linear analyses are discussed in subsequent sections.

2.3 Weight Calculations

For practical reasons, 4 in. wide sheet metal pieces of varying thicknesses were selected for columns and beams. These pieces are connected together through steel angles and bolted connections. Varying the thickness of the elements created the desired modes of failure. All structures had to fit within table weight limitations while being "weak" enough for the shaking table to push them into their non-linear range. Based on preliminary elastic analyses, tentative structural dimensions were determined for all three structures. Tables 5 and 6 present the final dimensions for structures fitting the three cases and the outlined constraints.

Low Rise Structure	**1**		**2**	
Column Plate				
Length	12.00	in	12.00	in
Width	4.00	in	4.00	in
Thickness	0.05	in	0.06	in
Weight	0.65	lb	0.85	lb
Beam Plate				
Length	8.00	in	8.00	in
Width	4.00	in	4.00	in
Thickness	0.11	in	0.05	in
Weight	0.96	lb	0.44	lb
Story Weight	2.26	lb	2.14	lb
No. of Stories	4.00		4.00	
Total Weight of Structure	9.04	lb	8.56	lb
	4.11	kg	3.89	kg
Additional Weights	0.75	kg	0.75	kg
20 % connections	1.42	kg	1.38	kg
Total Weight	**8.53**	kg	**8.27**	kg

Table 5: Weight Calculations for Structures 1 and 2

High Rise Structure	**3**	
Column Plate		
Length	6.00	in
Width	4.00	in
Thickness	0.02	in
Weight	0.15	lb
Beam Plate		
Length	4.00	in
Width	4.00	in
Thickness	0.10	in
Weight	0.44	lb
Story Weight	0.74	lb
No. of Stories	10.00	
Total Weight of Structure	7.36	lb
	3.35	kg
Additional Weights	0.30	kg
20 % connections	0.63	kg
Total Weight	**6.98**	kg

Table 6: Weight Calculations for Structure 3

Only Structure 1 was built as part of this project and will be discussed in detail. This modular structure with a Weak-Column Strong-Beam design was selected to achieve plastic hinging in the columns and illustrate the effect of column hinging on the overall behavior of the structure. The other two structures can be built in the future using similar details as those of Structure 1.

2.4 Details of Structure 1

Figure 12 shows a picture of the completed Structure 1 placed on the shaking table. Figures 13 and 14 show detailed drawings of the structure. All the connections of the structure are bolted so for ease of assembly and disassembly. If elements are damaged during shaking, they can easily be removed and replaced with new ones.

Figure 12: Modular Structure

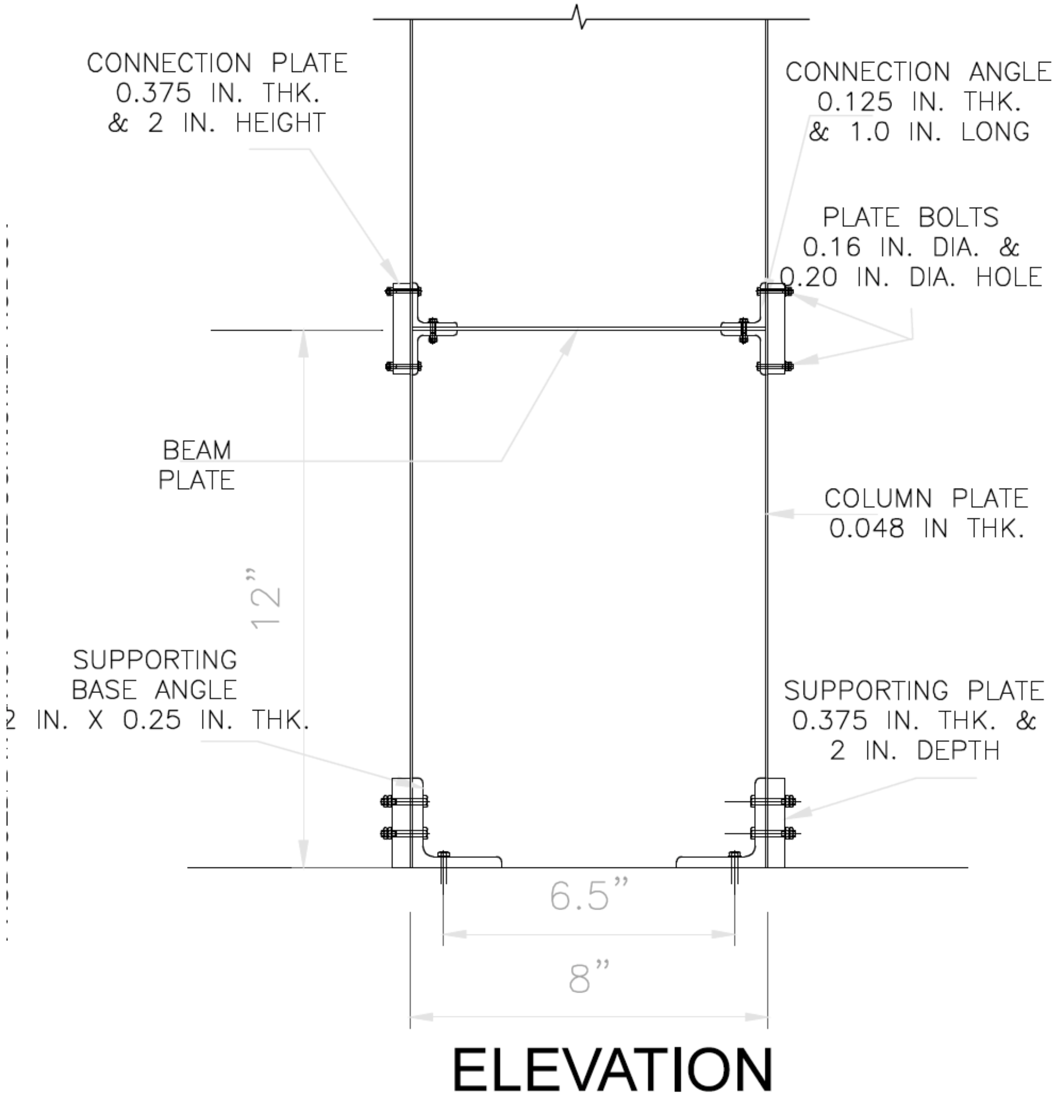
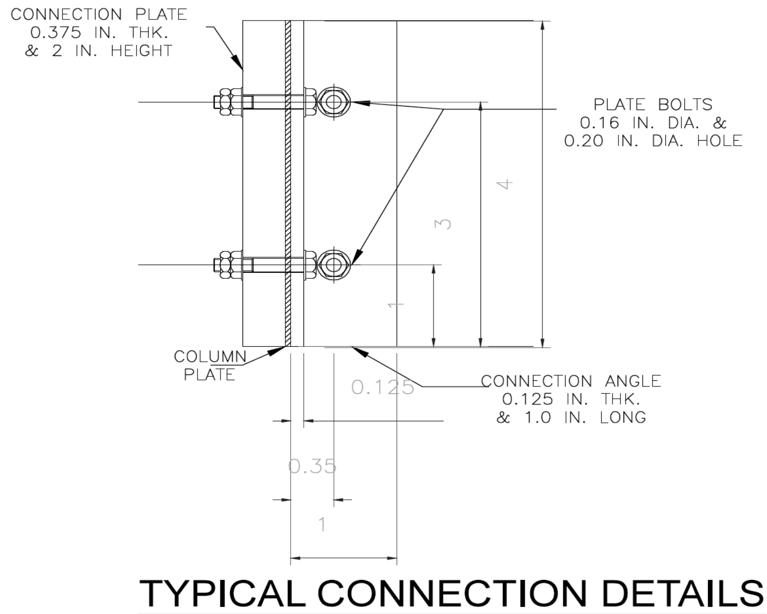


Figure 13: Elevation of the Modular Structure



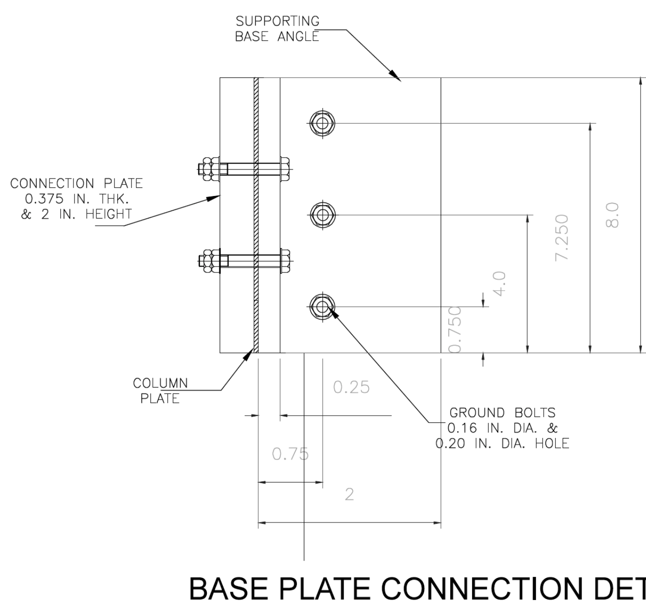


Figure 14: Sectional details of the Modular Structure (dimensions in in.)

2.5 Non-Linear Analyses of the Structure

Three different types of analyses were carried out. An Eigen-value analysis was performed to determine the natural periods and mode shapes of the structure. A pushover analysis was performed using a first-mode distribution of vertical forces to identify critical behavioral milestones. Lastly, sample ground motions were applied to the structure and its response under the excitations studied. OpenSEES input files are included in Appendix.

2.5.1 OpenSEES

OpenSEES stands for "Open System of Earthquake Engineering Simulation"[4]. It is an open-source analytical software developed for simulating the seismic response of structures. OpenSEES provides users with a wide range of material models, elements, and solution algorithms. It is a powerful tool for non-linear analysis as it can deal with material as well as geometric non-linearity very efficiently. It requires scripted input while it outputs text files with analysis results. The output requires post-processing in order to get figures or graphs. In short, OpenSEES is less user-friendly than commercial software but much more powerful and transparent.

2.5.2 Material

The OpenSEES steel material "Steel02" in was used to model all elements. The uniaxial stress-strain relation for this material follows the Giuffre-Menegotto-Pinto rules[5] with isotropic strain hardening. Figure 15 illustrates the stress-strain relation for monotonic loading. The strain-hardening ratio that is the ratio between post-yield modulus and initial elastic modulus was taken as 0.01 for all elements. The yield stress was specified at 36,000 psi, for A36 type steel.

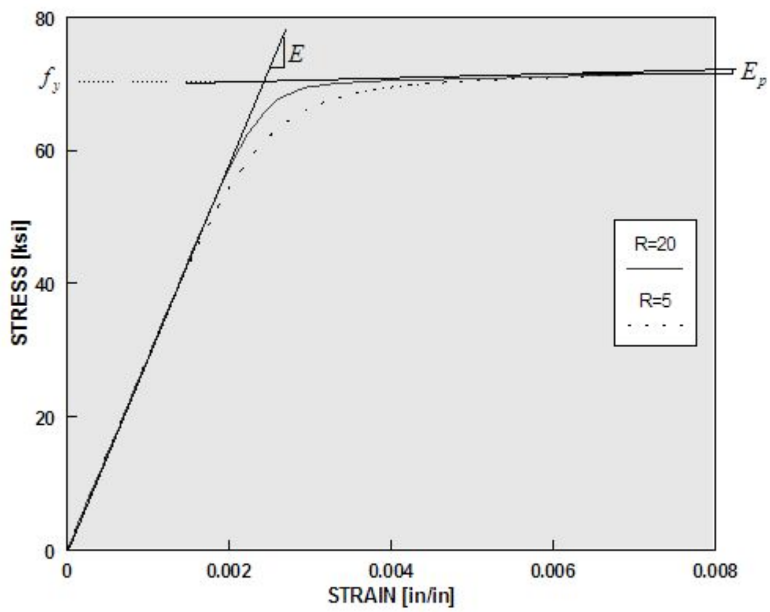


Figure 15: Steel02 Stress-Strain Monotonic Relation[4]

2.5.3 Element

Force-based beam-column elements[6] were used to model all beams and columns. Each beam and column was subdivided into three elements. Two end elements with length equal to twice the element thickness (in direction of loading) and a third element bridging the two. End elements had two integration points while mid element had five integration points. Plasticity spread was constrained to end elements. Element integration is based on the Gauss-Lobatto quadrature rule. Joints were modeled as rigid elements. Figure 16 presents a schematic drawing of the analytical model of Structure 1.

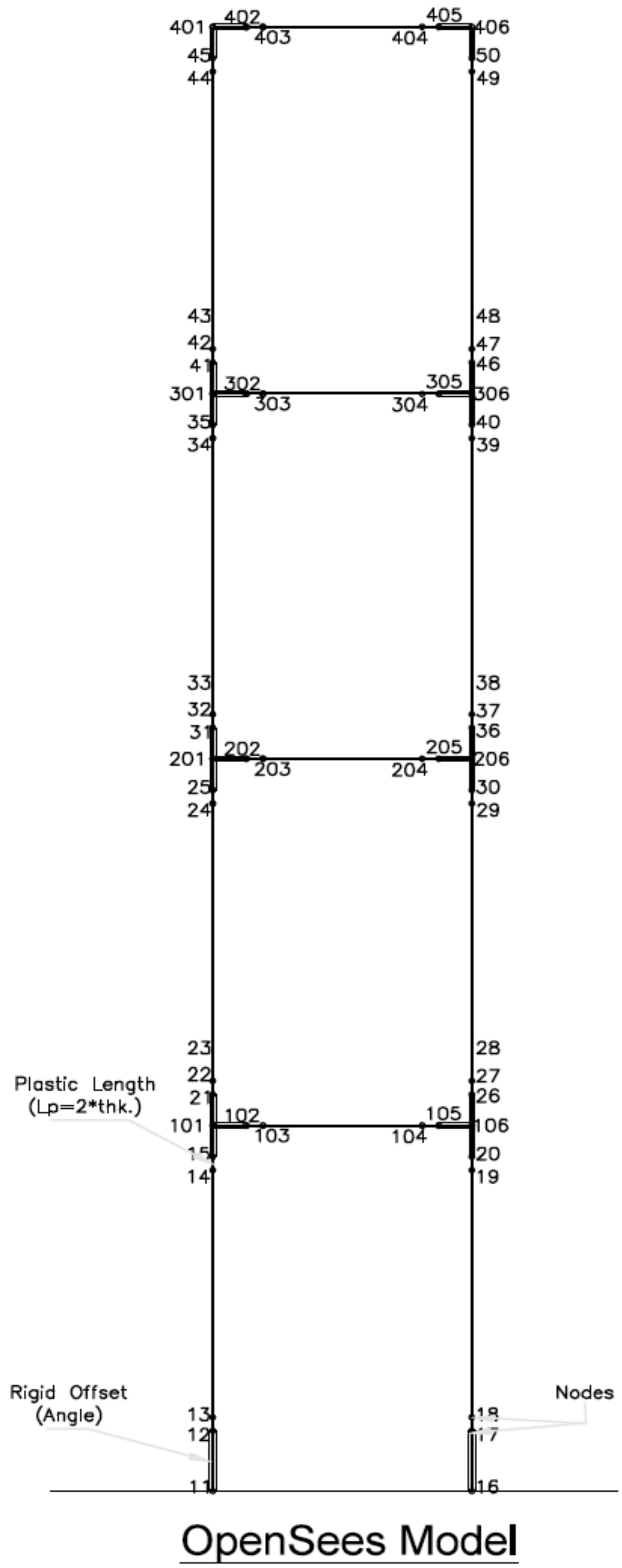


Figure 16: Nodal Geometry used in the OpenSEES[4] model

2.6 Pushover Curve

The structure was pushed until the top story (i.e. node 401) was displaced laterally to 20 in. by increments of 0.1 in.. A lateral load pattern representing the first mode of the structure was applied for the pushover analysis. That load pattern was represented by a 2 lb force at the first story and increased by 2 lbs per floor until it was 8 lbs at the top floor. The pushover curve is shown in Figure 17.

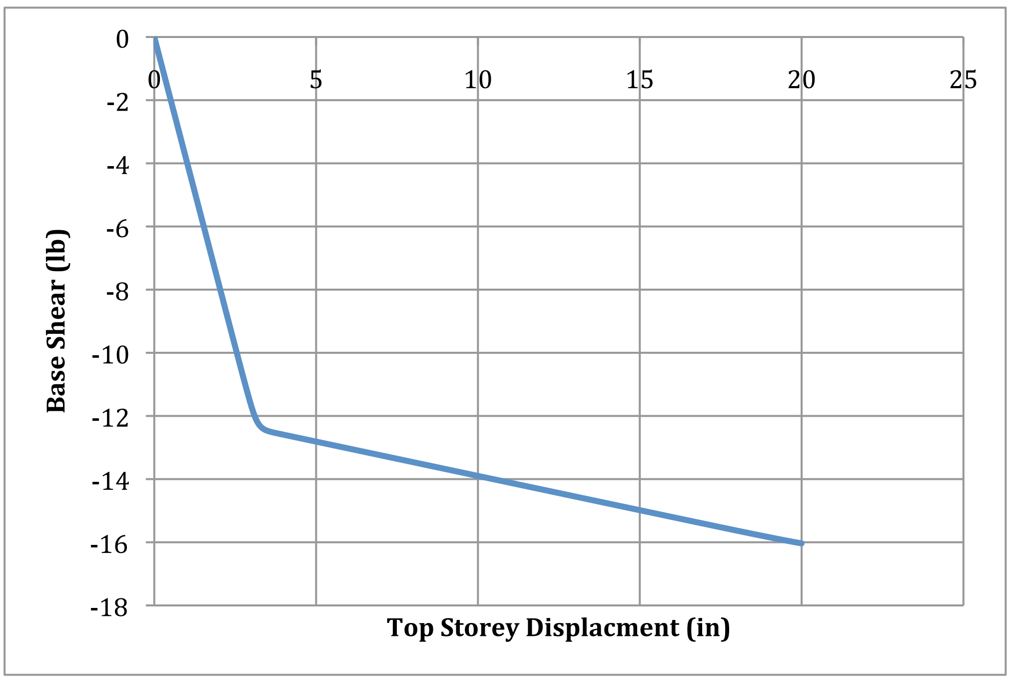


Figure 17: Pushover Curve

2.7 Mode Shapes

An Eigen-value analysis is carried out to determine first four natural modes of the structure. First four mode shapes, along with the natural periods are shown in Figure 18.

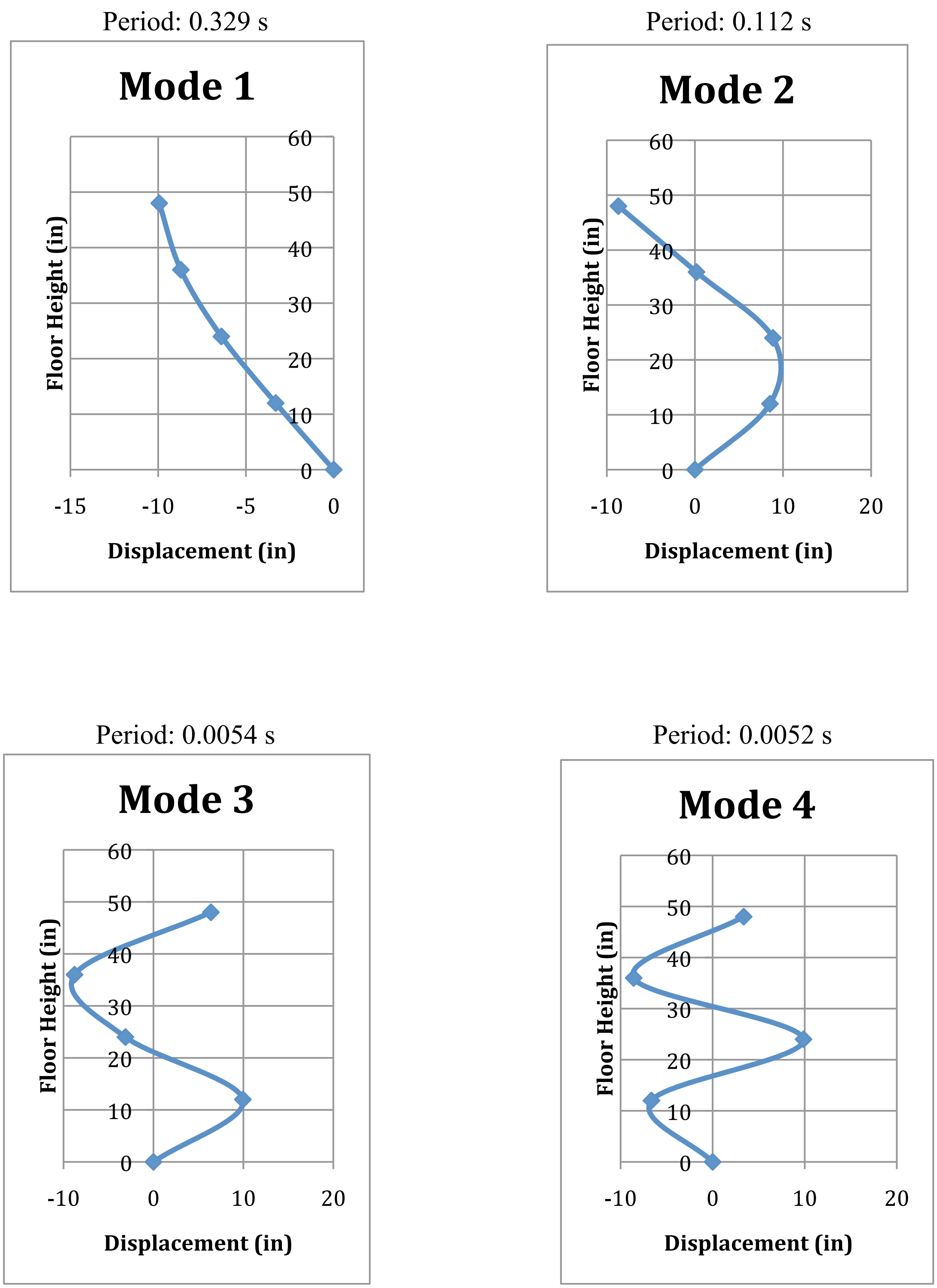


Figure 18: Mode Shapes of the Structure

2.8 Response for Kobe Ground Motion

Acceleration record for station Takatori in 00 direction during Kobe earthquake is used in the OpenSEES analysis. For the recorded acceleration values, plastic hinge formation is observed in the end elements of the first story columns. The hinging mechanism can be seen in Figures 19, 23 and 24. For the upper floors, the response is elastic, which can be observed from figures 20, 21, 22, 25, 26, 27 and 28. As expected, yielding phenomenon is observed only at the column ends of the first story, and all the beams in the structure remain elastic. This behavior represents the weak column strong beam response. All the graphs included in this section are obtained by considering only material non-linearity. P-Delta, i.e geometric non-linearity is discussed in the succeeding section.

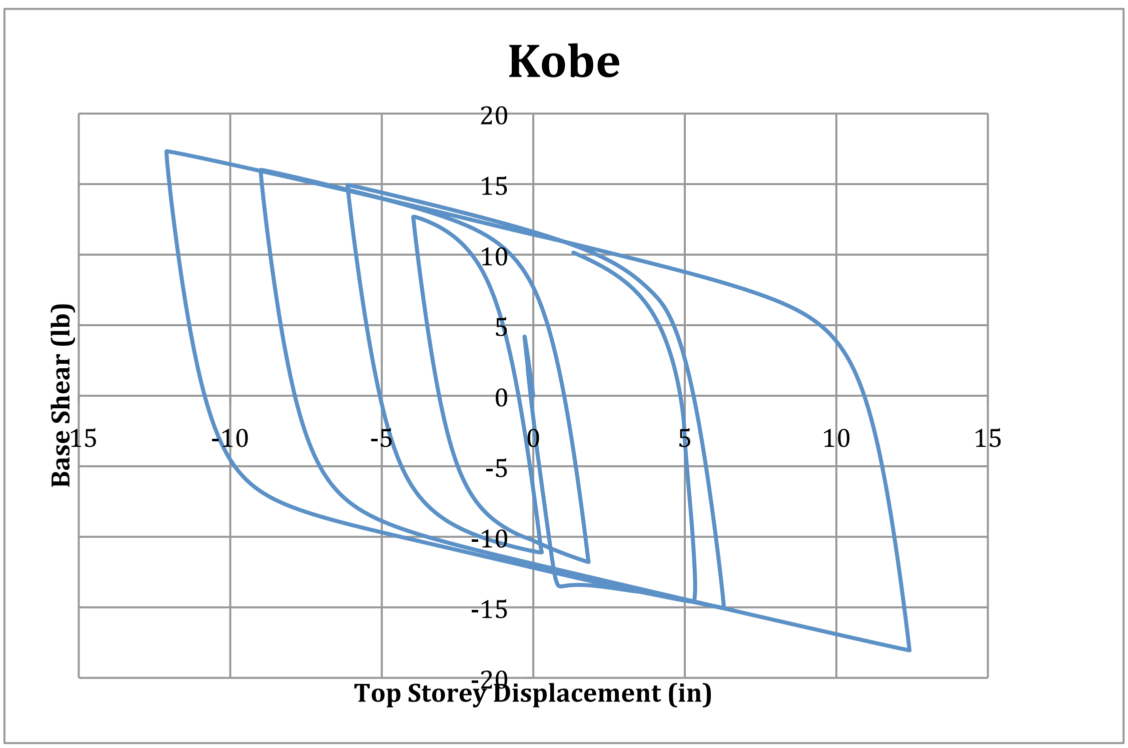


Figure 19: First Story Column

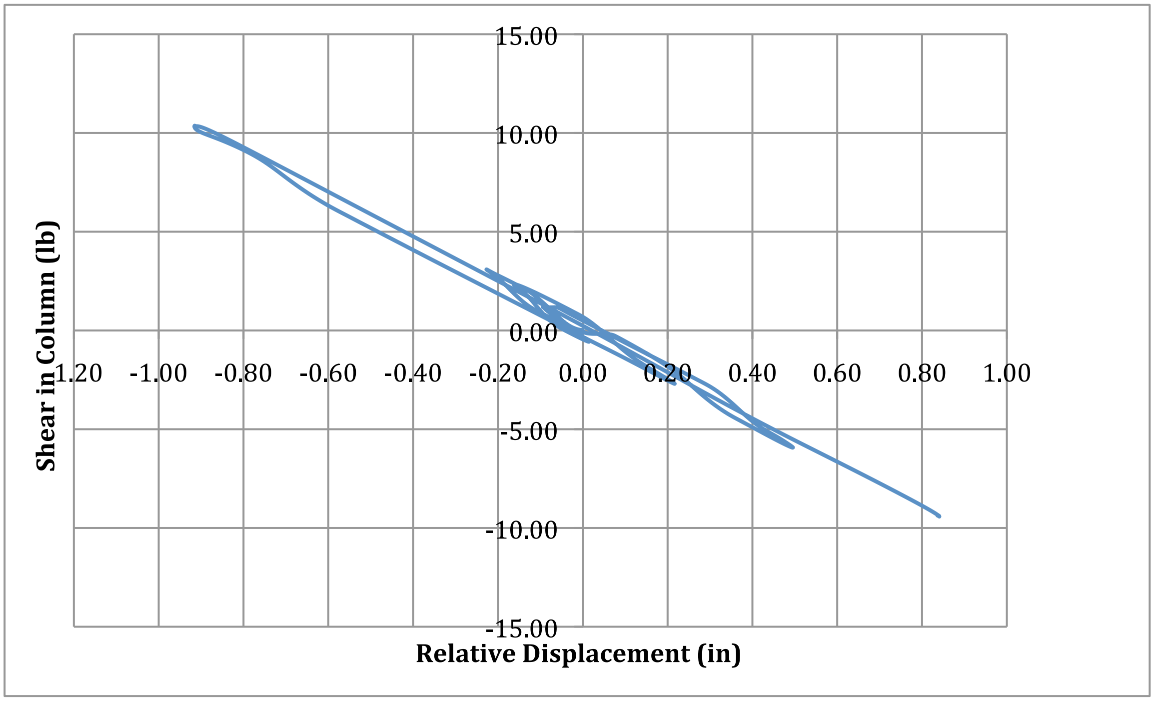


Figure 20: Second Story Column

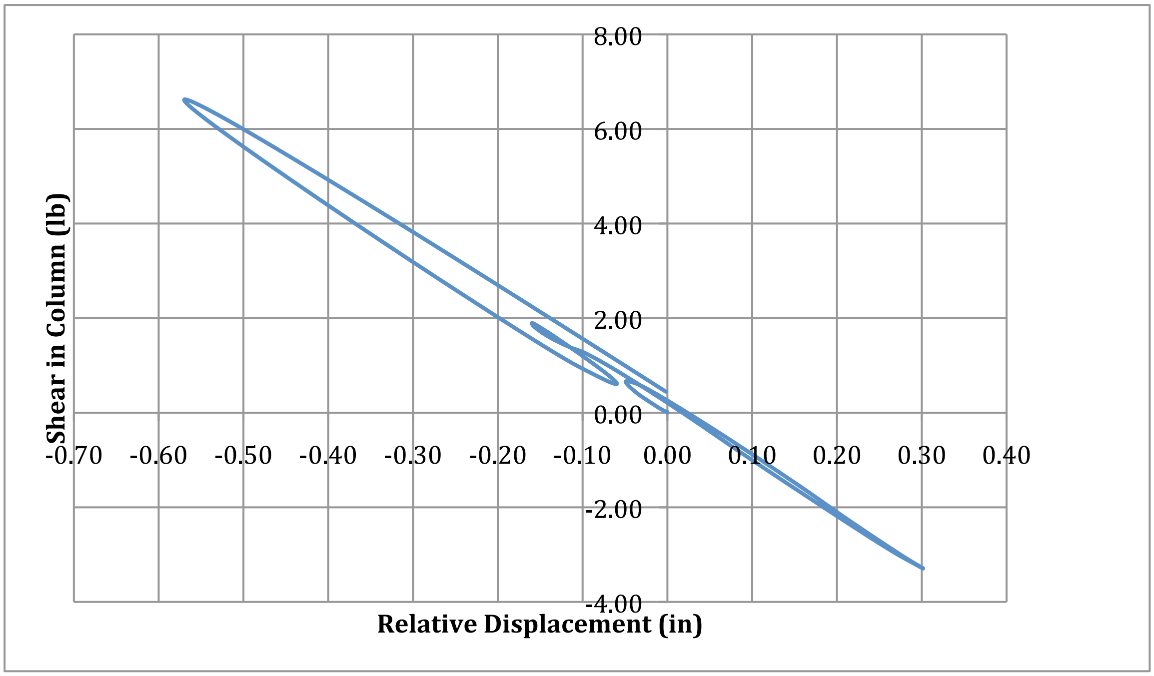


Figure 21: Third Story Column

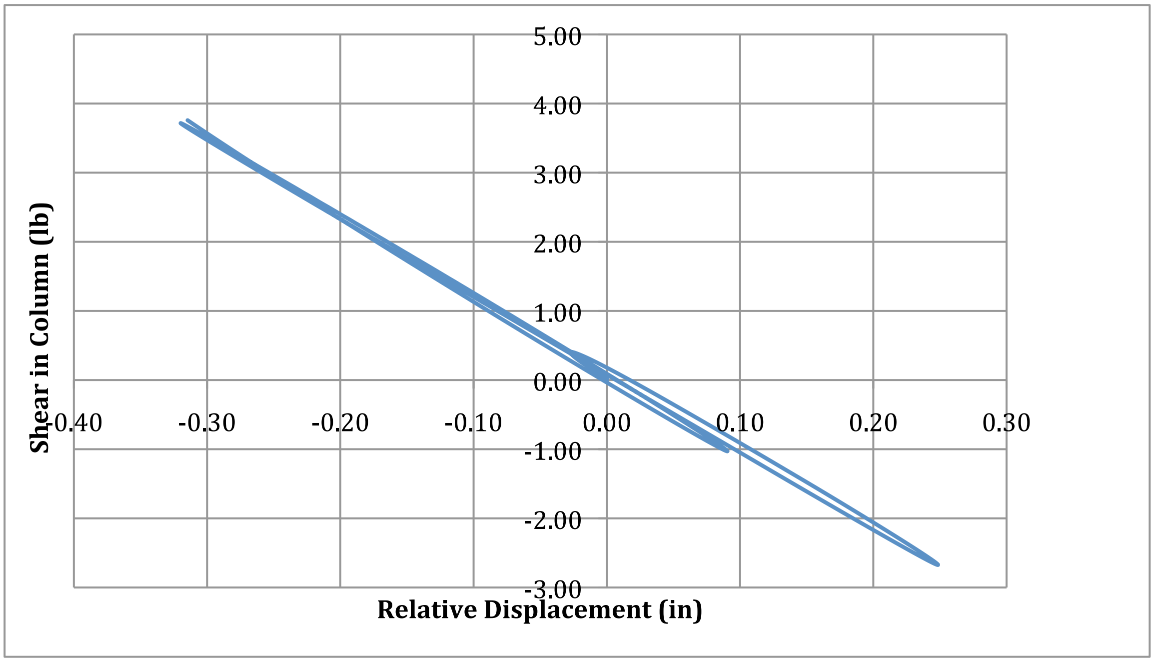


Figure 22: Forth Story Column

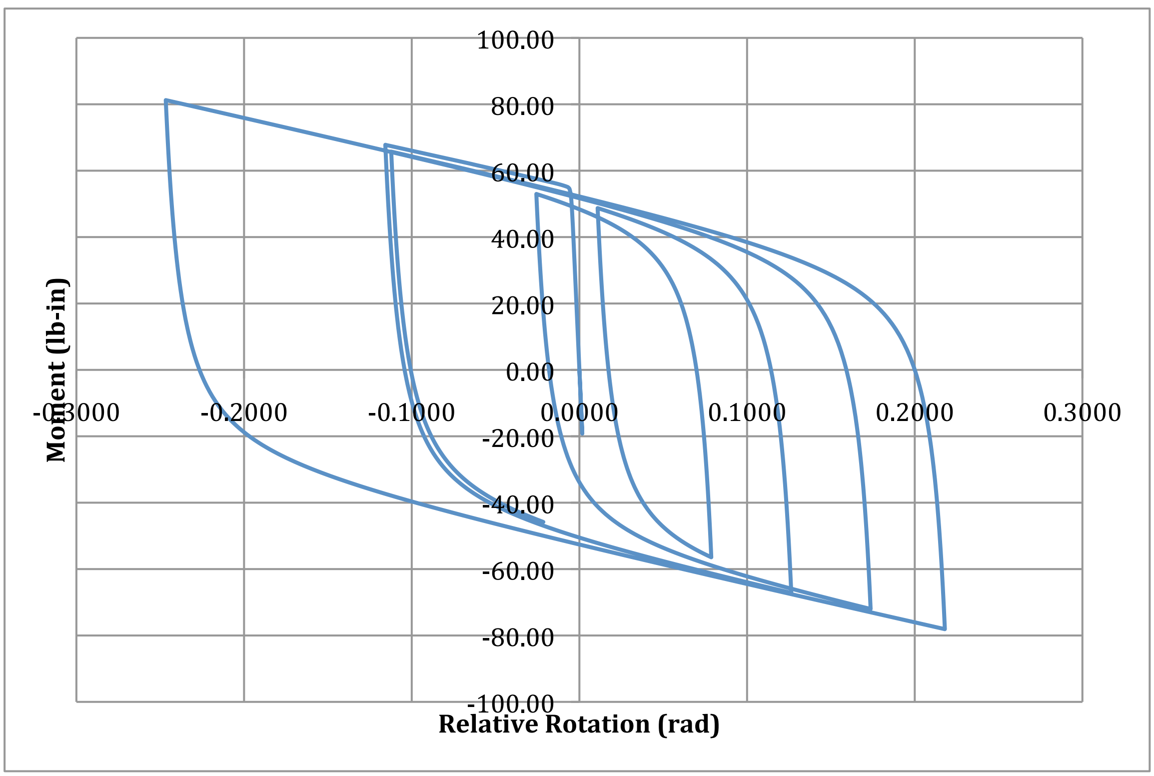


Figure 23: First Story Column - Bottom End

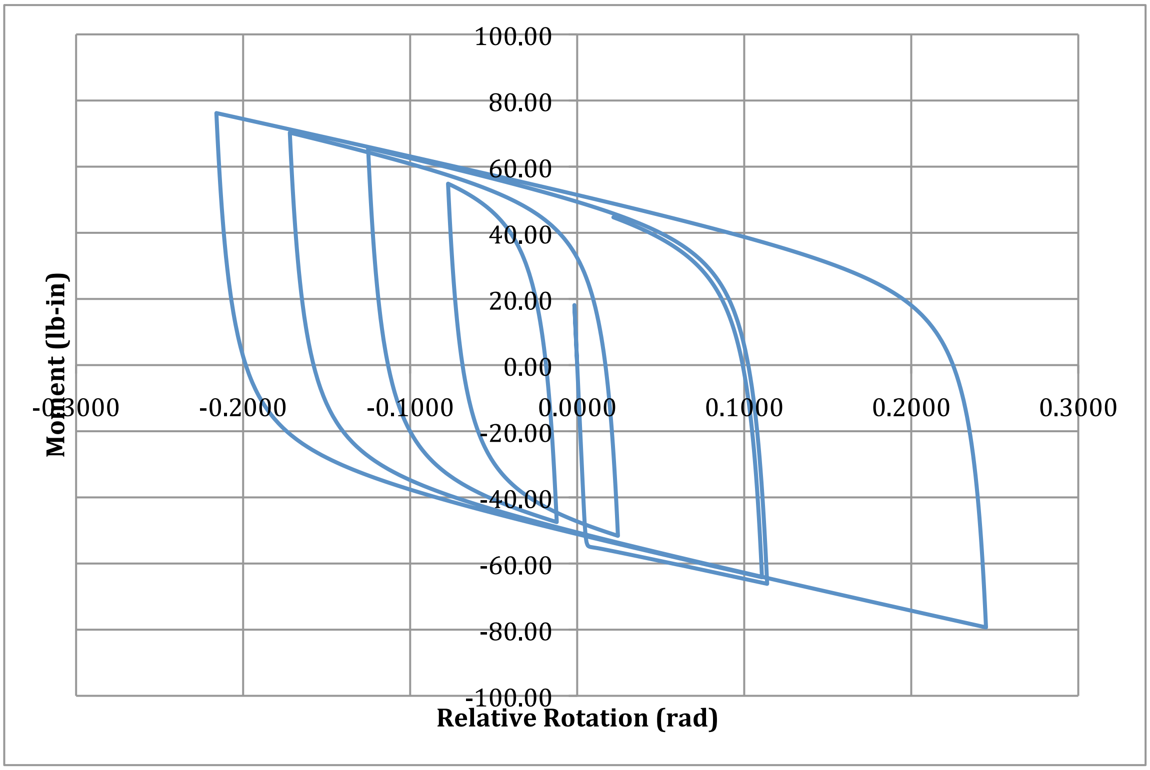


Figure 24: First Story Column - Top End

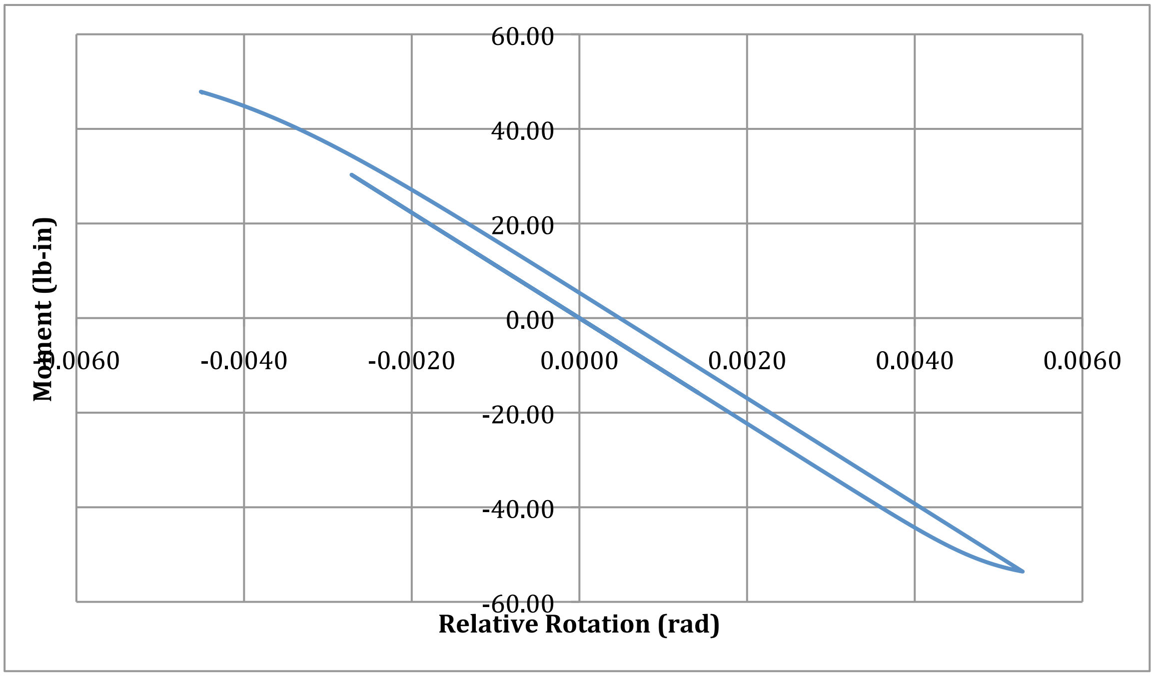


Figure 25: Second Story Column

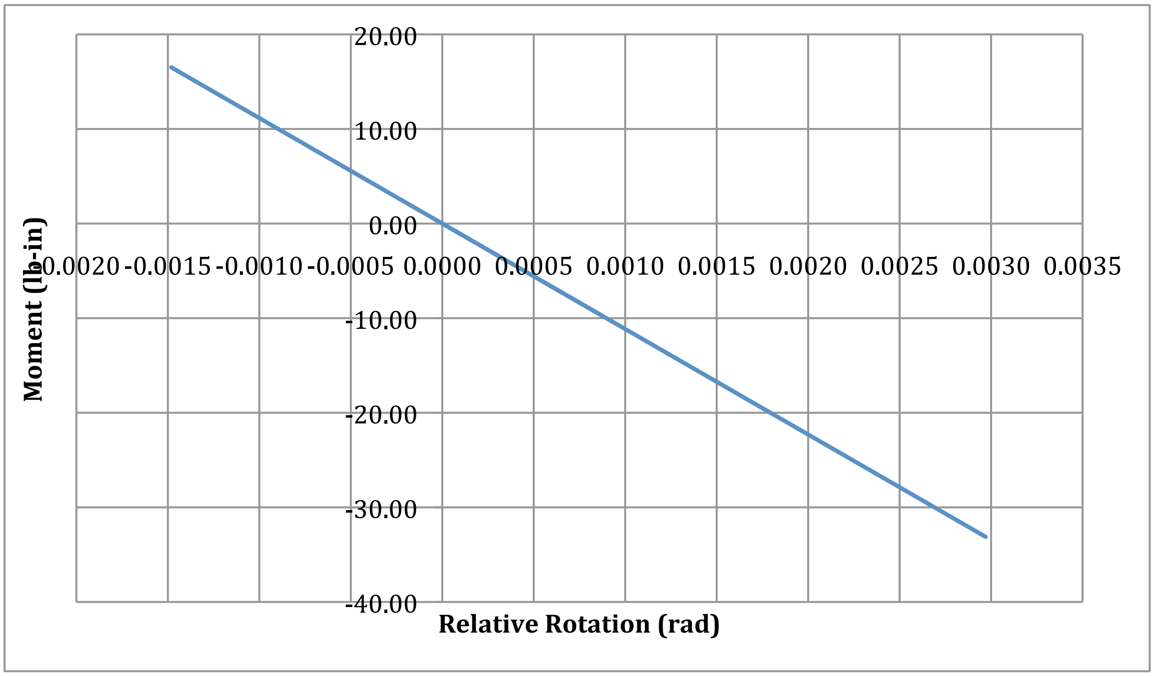


Figure 26: Third Story Column

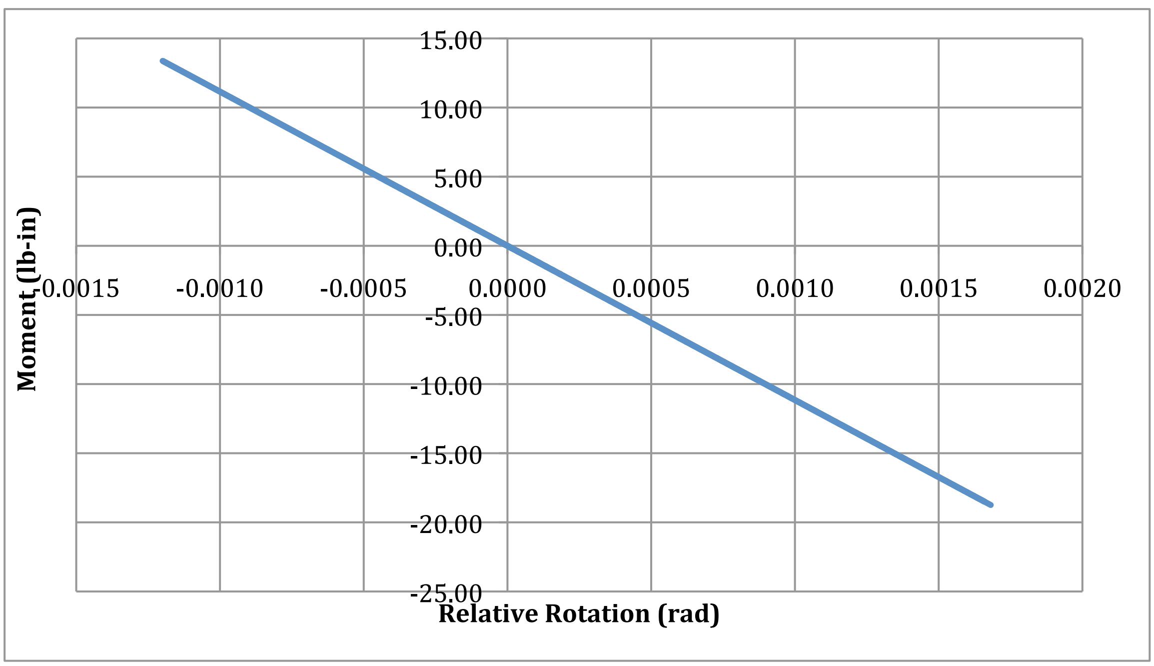


Figure 27: Fourth Story Column

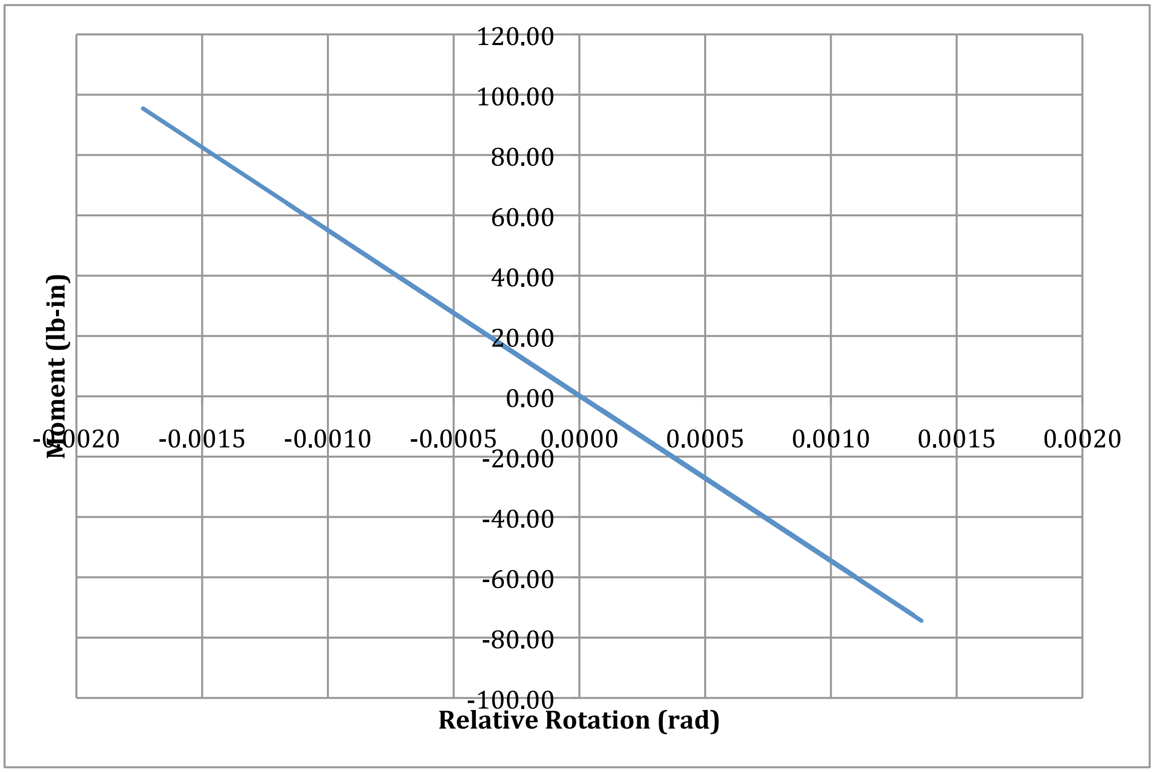


Figure 28: First Story Beam

2.9 Response for Northridge Ground Motion

Acceleration record for station Castaic Old Ridge RT in 90 direction during Northridge earthquake is used in the OpenSEES analysis. For the recorded acceleration values, plastic hinge formation is observed in the end elements of the first story column. The hinging mechanism can be seen in figures 29, 33 and 34. For the upper floors, the response is elastic, which can be observed from figures 30, 31, 32, 35, 36, 37 and 38. As expected, yielding phenomenon is observed only at the column ends of the first story, and all the beams in the structure remain elastic. This behavior represents the weak column strong beam response. All the graphs included in this section are obtained by considering only material non-linearity. P-Delta, i.e geometric non-linearity is discussed in the succeeding section.

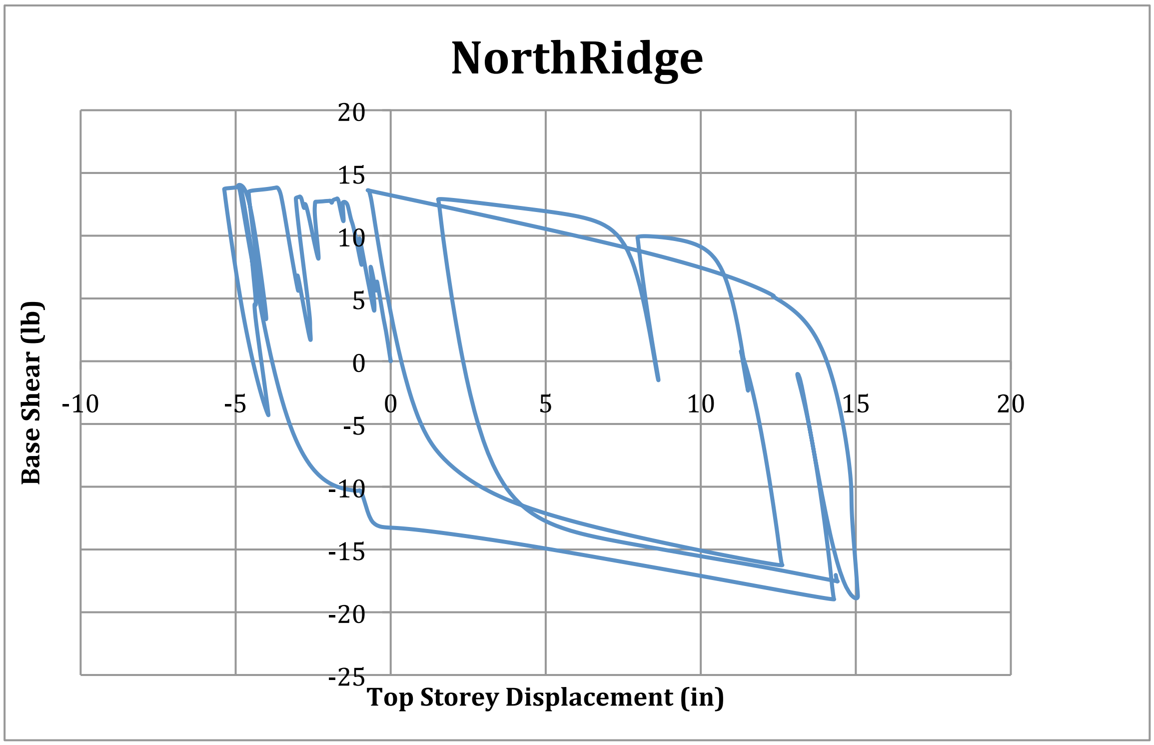


Figure 29: First Story Column - Bottom End

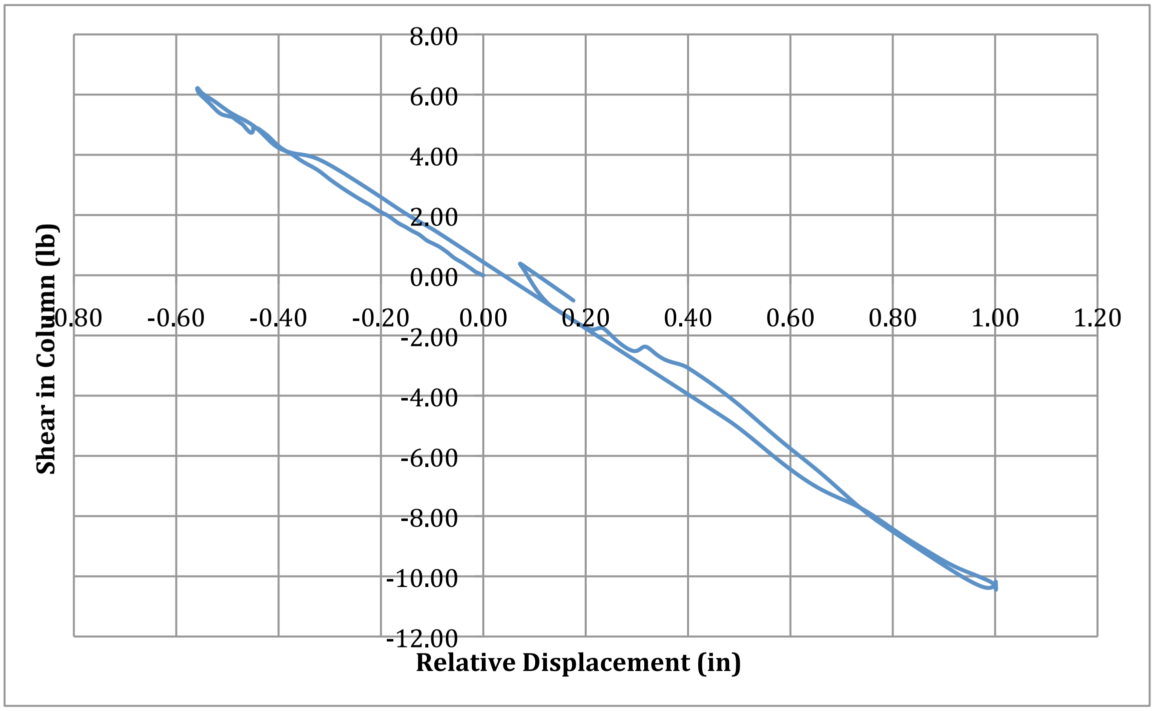


Figure 30: Second Story Column

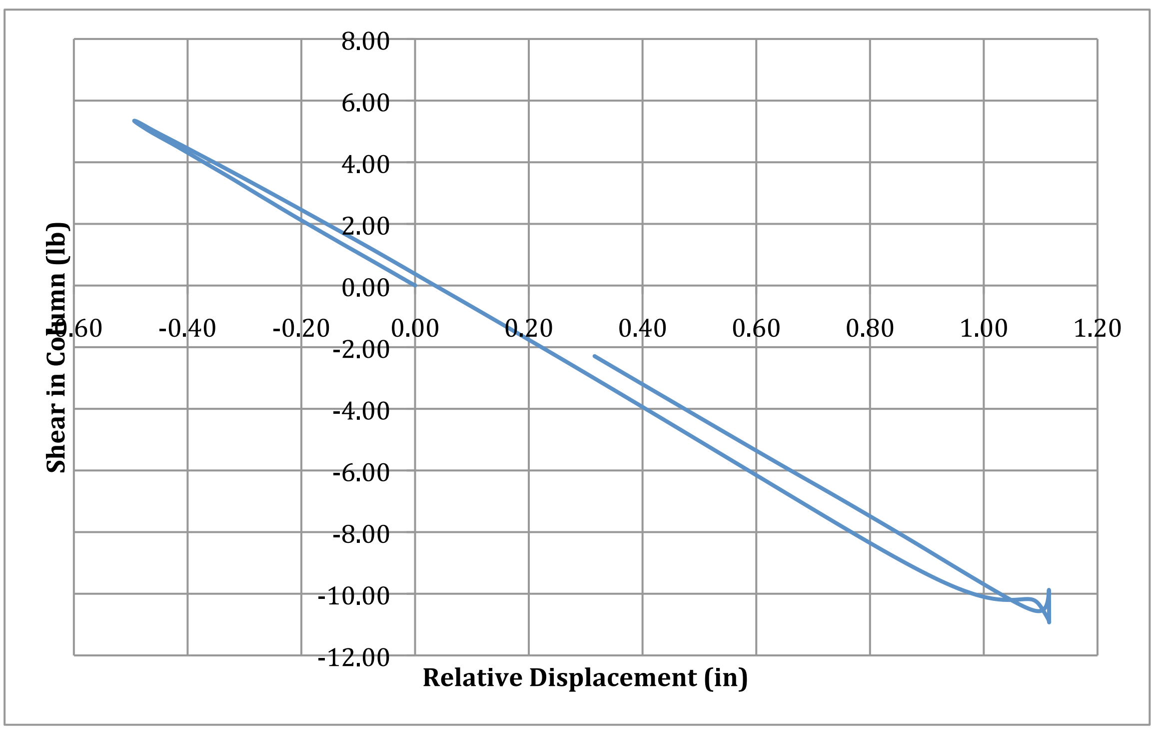


Figure 31: Third Story Column

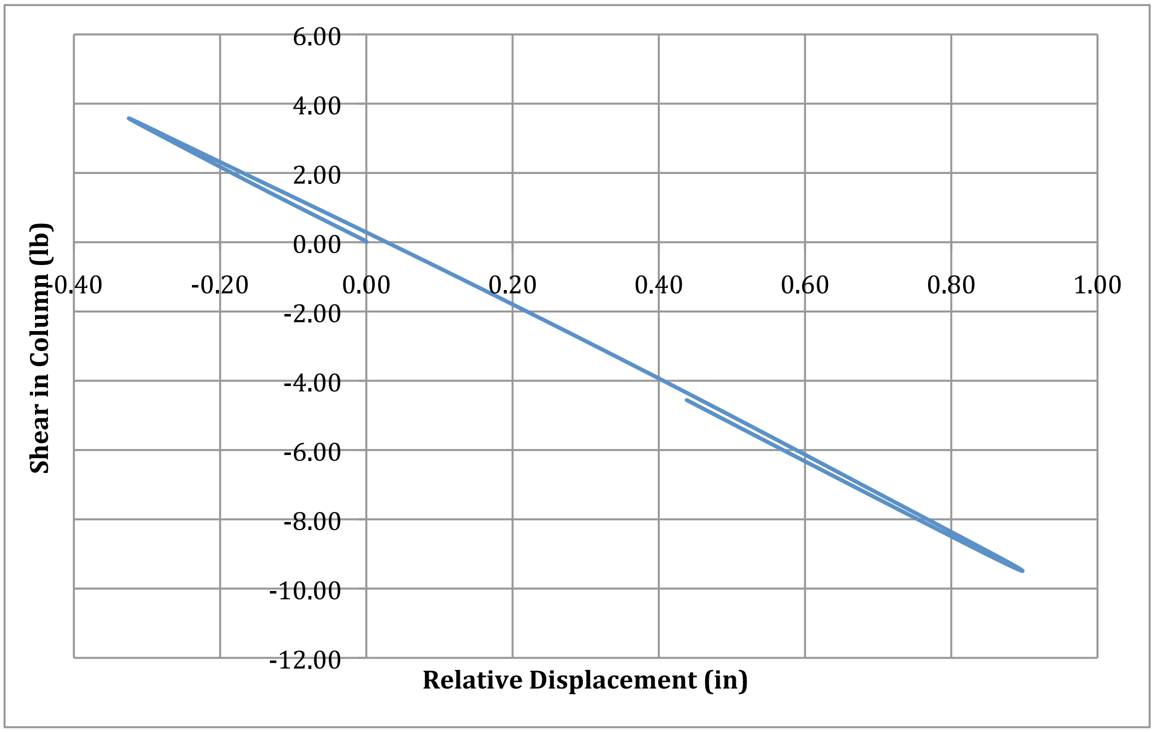


Figure 32: Fourth Story Column

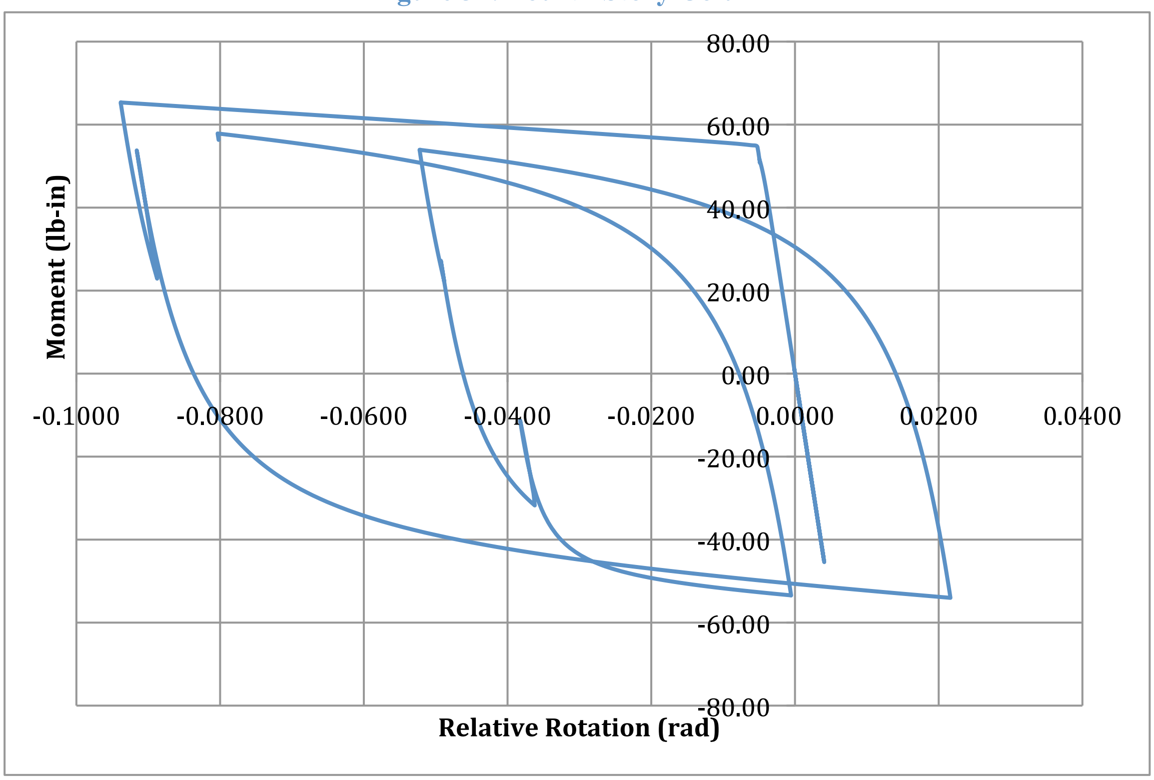


Figure 33: First Story Column - Bottom End

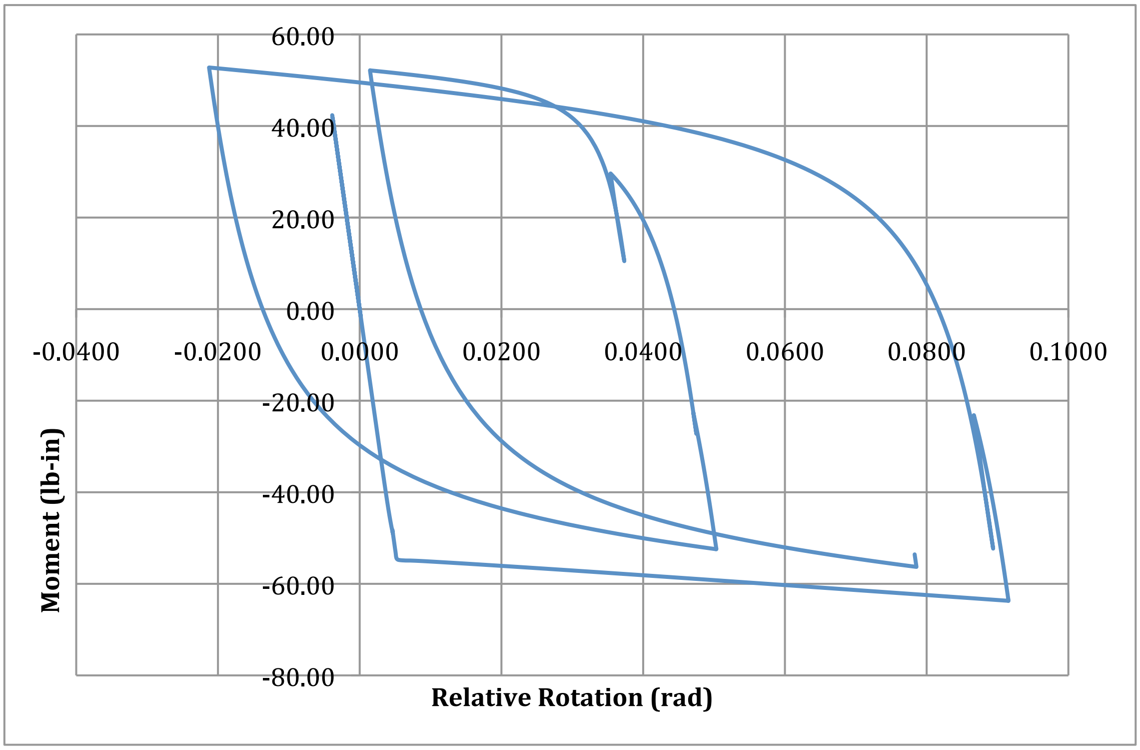


Figure 34: First Story Column - Top End

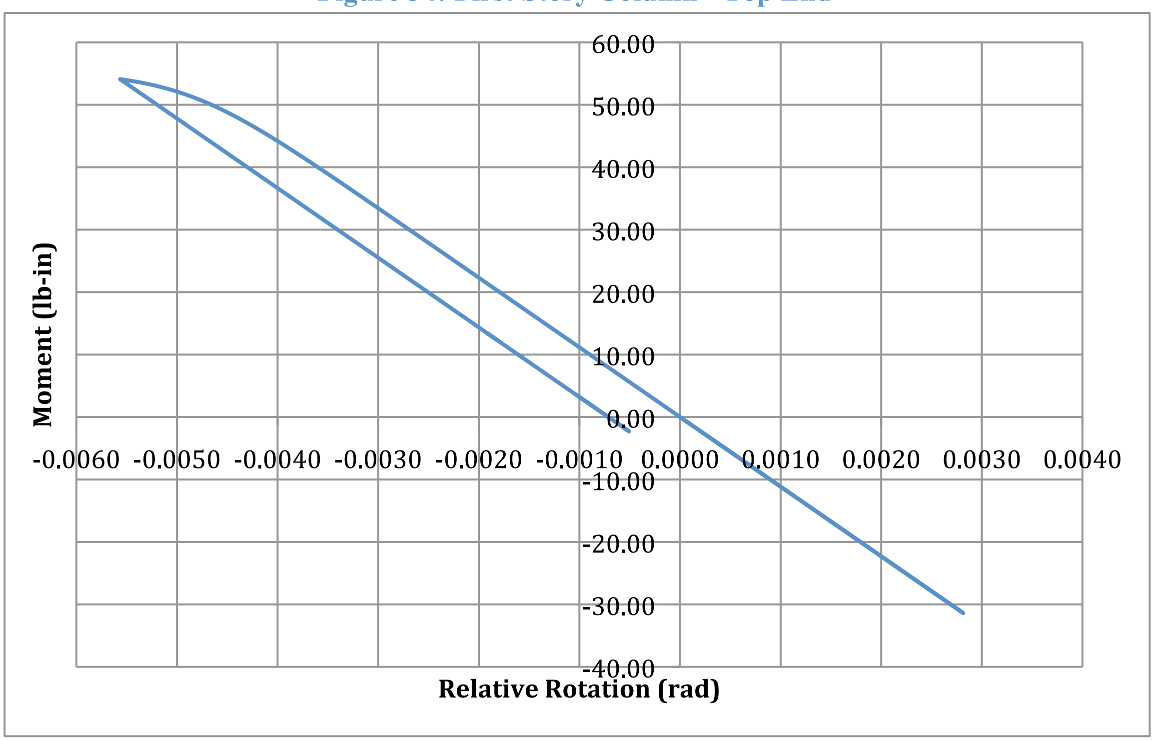


Figure 35: Second Story Column

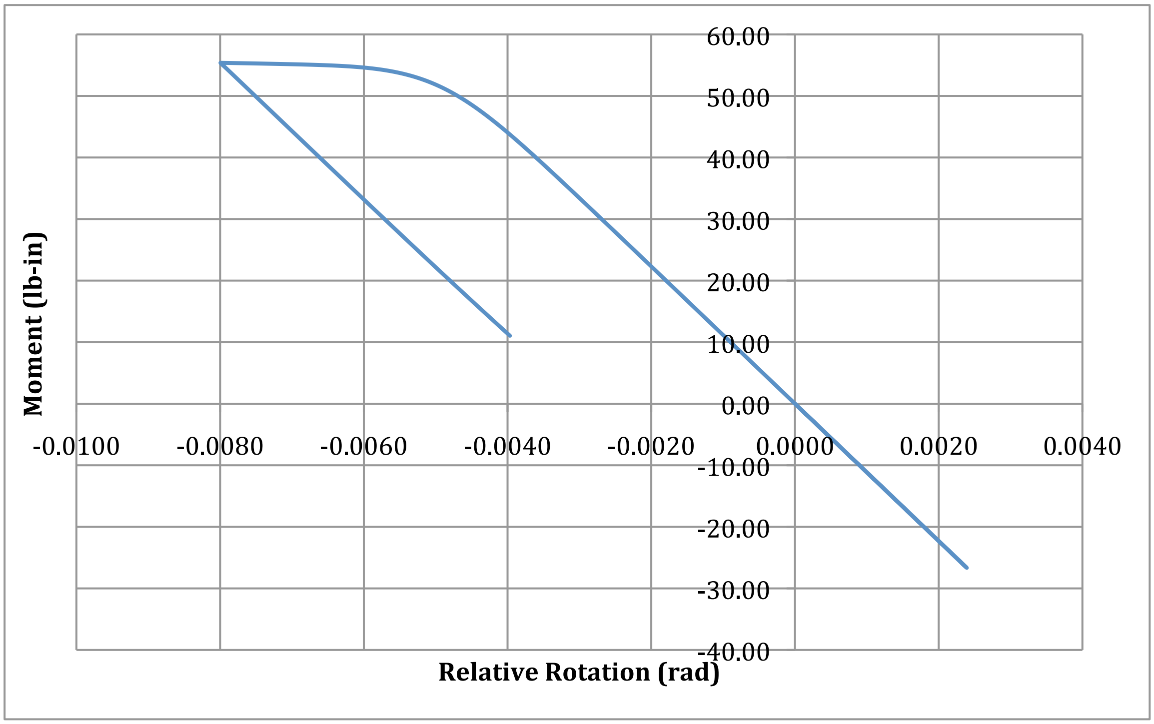


Figure 36: Third Story Column

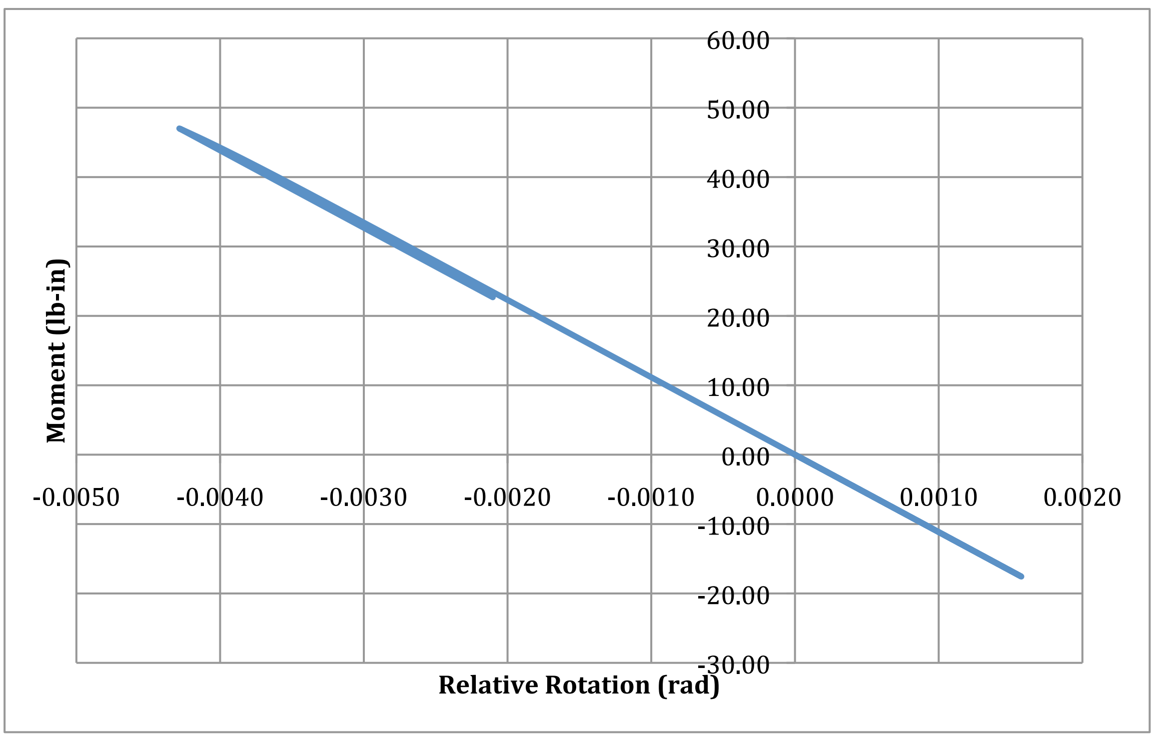


Figure 37: Fourth Story Column

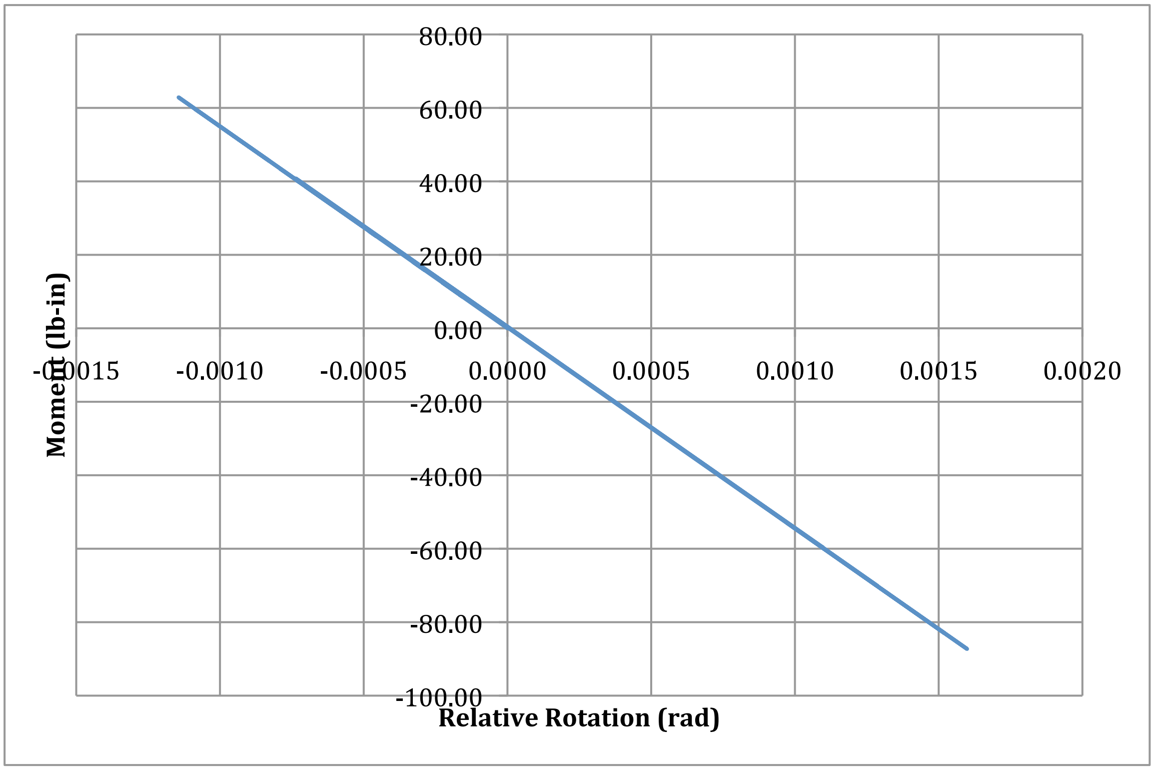


Figure 38: First Floor Beam

2.10 Geometric Non-Linearity (or P-Delta) Effects – Kobe Ground Motion

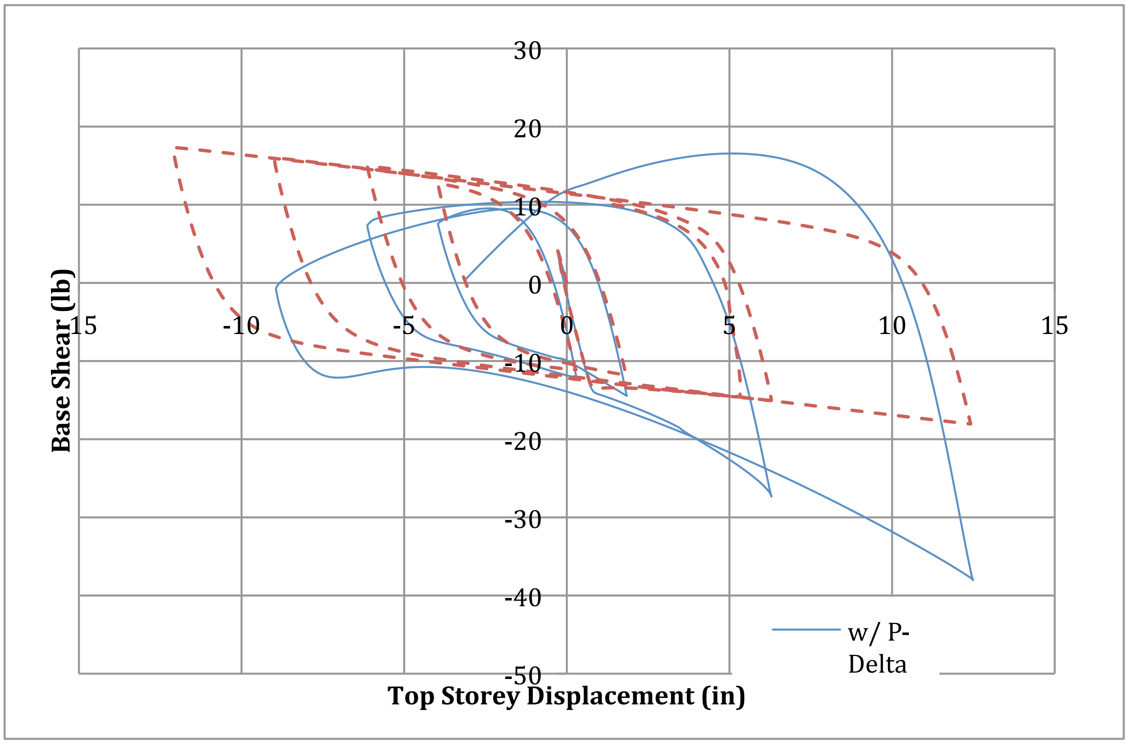


Figure 39: Geometric Non-linearity Effects on the Structure

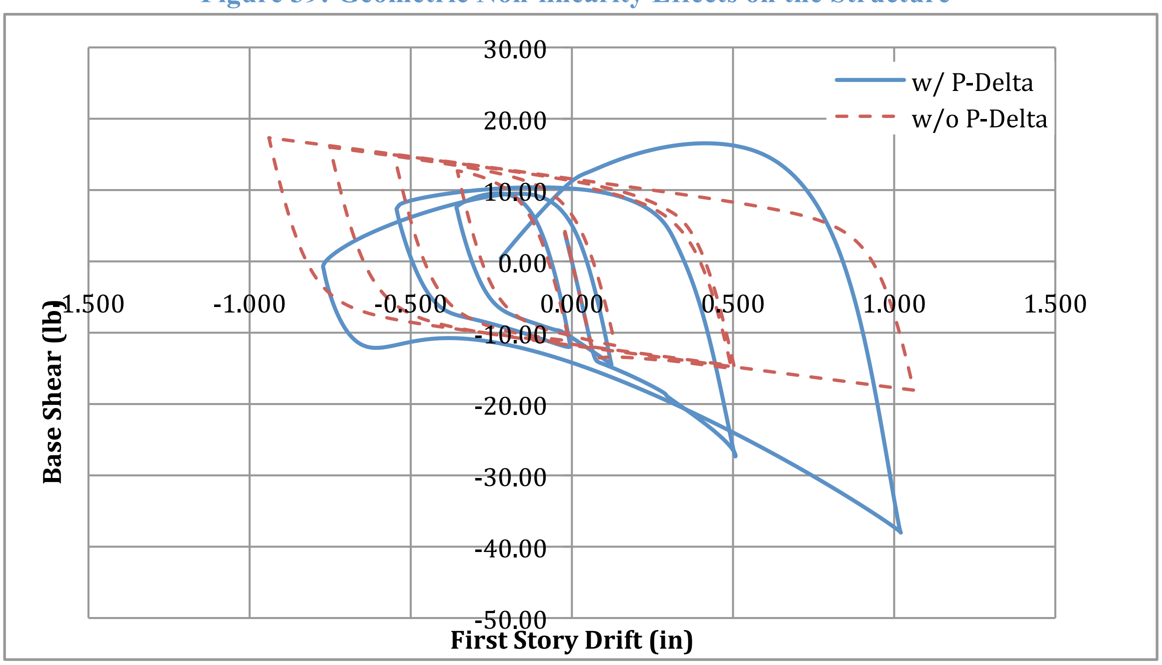


Figure 40: Geometric Non-linearity Effects on the Structure

Geometric non-linearity was taken into account in the analysis to estimate its influence on the response. The two responses with and without P-Delta effects are compared in Figures 39 and 40. Figure 39 shows the graph of base shear vs. top story displacement of the structure, while Figure 40 illustrates the variation of base shear with the first story drift. Due to the flexibility of the structure, geometric non-linearity affected the response significantly. It was found that the effect is more pronounced and detrimental when the element under consideration is in compression.

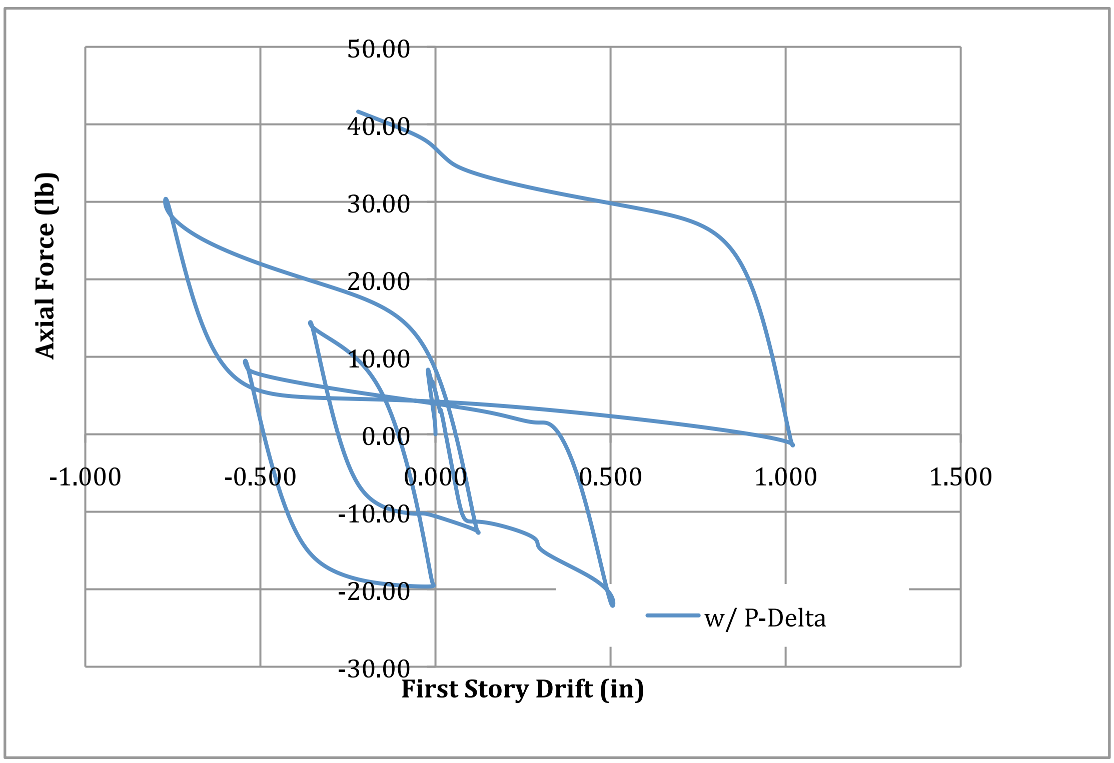


Figure 41: First Story Left Column

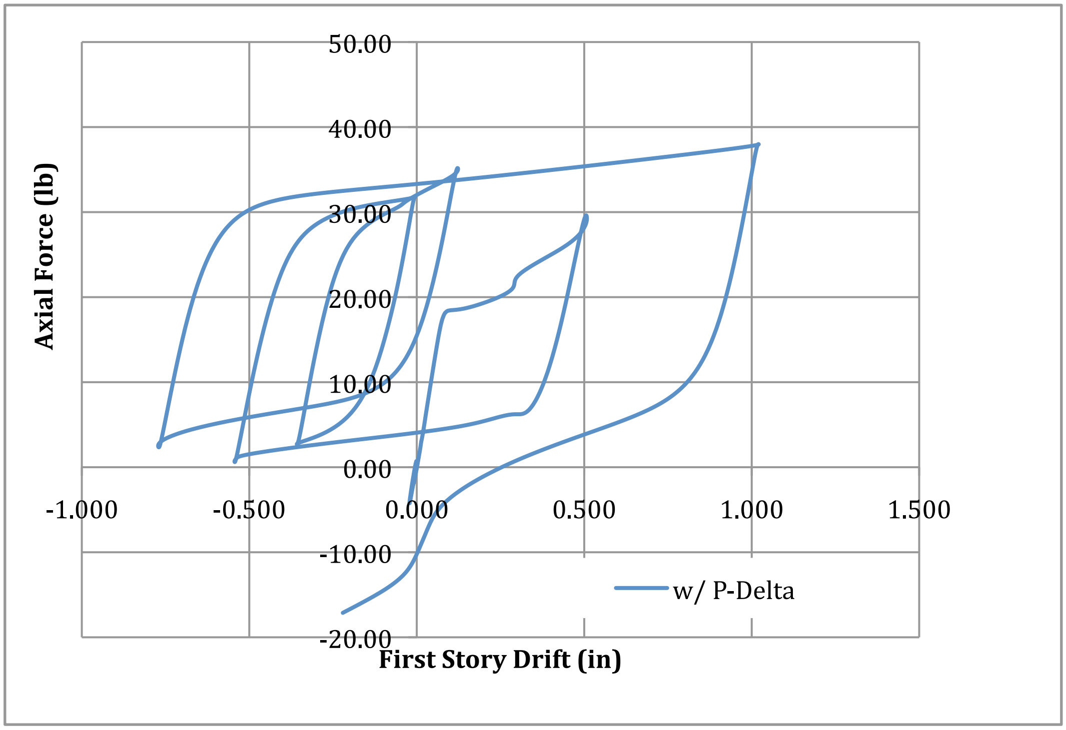


Figure 42: First Story Right Column

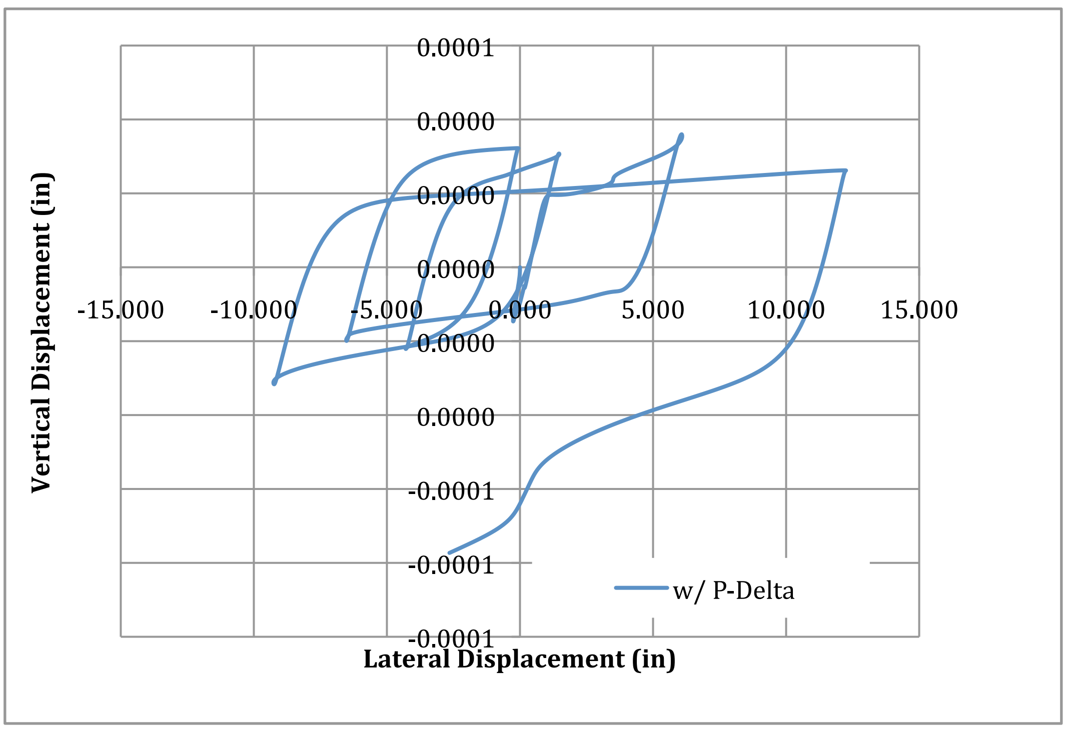


Figure 43: Left Beam-Column Joint at the First Floor Level

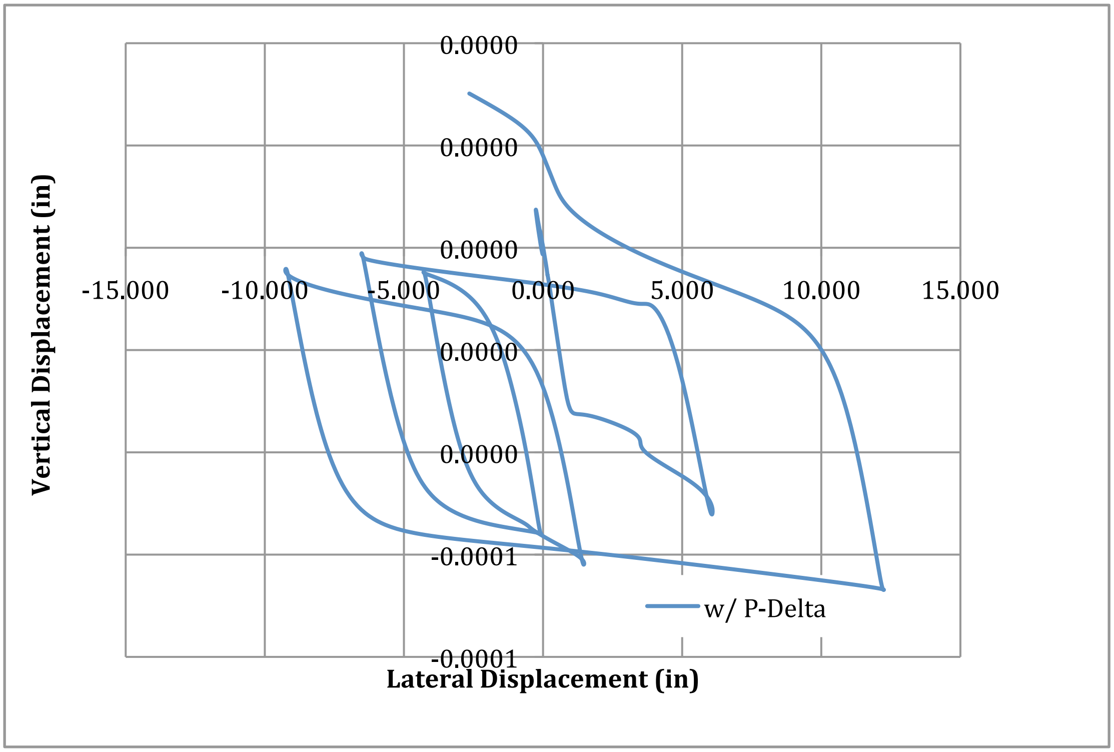


Figure 44: Right Beam-Column Joint at First Floor Level

The Figures 41 and 42 show the variation of axial force with the first story drift for the left and right column of the first story respectively. The extent of lateral and vertical displacements of the left and right joints at the first floor level during the ground motion is shown in the Figures 43 and 44 respectively. It can be seen that the vertical displacements are negligible as compared to horizontal displacements. This can be accounted to the nature of the ground motion and also the flexibility of the column sections.

2.11 Proofing: Test Modules

The modular structure is mounted on the shaking table and tested using various input signals available such as sine wave; sweep wave and recorded ground motions. The testing procedure can be divided into following modules so that the change in response of the structure can be observed and studied.

1. Application of sine wave: Modal shapes and periods
2. Application of default ground motions in simple program: Response of the structure to Northridge and Kobe ground motions
3. Application of ground motions: Response of the structure to user input ground motions to demonstrate yielding of first story column plates.

2.11.1 Module 1: Application of Sine Wave

Purpose: To study the modal shapes of the structure using a user-defined sine wave.

Procedure:

1. A sine wave can be applied to the structure either using the simplified Quanser Shake Table II Software or by using the QuaRC Controllers.
2. The graphical user interface can be used as per section 1.7.1, which enables the user to change the amplitude and frequency of the sine wave.

3. The QuaRC Controllers can also be used as per section 1.9, which enables the user to set the amplitude and frequency through Matlab-Simulink scripts.
4. Vary the sine wave frequency to achieve resonance behavior at each fundamental period. At resonance the modes of the structure can be observed.

Observations:

Description	**Frequency (Hz)**
Mode 1	
Mode 2	
Mode 3	
Mode 4	

Table 7: Mode Shapes of the Structure

2.11.2 Module 2: Application of Ground Motions

Purpose: To study the response of the structure to various recorded ground motions.

Procedure:

1. Default Kobe and Northridge ground motions can be applied using the Quanser Shake Table II Software as explained in section 1.7.1. or QuaRC controller can be used in order to apply any other recorded ground motion using Matlab-Simulink scripts as per section 1.7.2.
2. The default ground motions keep the structure in the linear range and there is no plastic hinge mechanism.

Observations:

Response of the structure on application of these ground motions can be studied by adding accelerometers or displacement transducers to the structure to evaluate its performance. This falls within future work.

2.11.3 Module 3: Formation of plastic hinge mechanism

Purpose: To study the yielding of structure on application of scaled ground motions.

Procedure:

1. QuaRC controller can be used in order to apply any other recorded ground motion using Matlab-Simulink scripts as per section 1.7.2.
2. The ground motions used in the OpenSEES analysis, like the one recorded at the Takatori Station during Kobe earthquake in the 00 direction, is capable of forming plastic hinge mechanism in the structure. While, the default ground motions need to be scaled to achieve yielding in the structure.
3. A ground motion scaling factor “scale_factor” is included in the make_quake.m MATLAB script. The user can modify this scaling factor, in order to scale up or scale down the recorded acceleration data.

Observation:

Potential hinging action is studied on application of scaled ground motions and yielding phenomenon in steel structures can be observed. Response of the structure on application of these ground motions can be studied by adding accelerometers or displacement transducers to the structure to evaluate its performance. This falls within future work.

3. Conclusion

The prime objective of the project was to provide a modular structure that can be tested on an educational shaking table to demonstrate basic concepts in structural dynamics and earthquake engineering. A scaled four story steel structure was built for that purpose. The structure is made up of locally available steel sections and has bolted connections, which makes it truly modular and easy to assemble or dissemble. The structure can be easily modified by changing the plate sections, story heights, or even number of stories in order to learn more about response of a multi-storied steel frame building to ground motions.

A summary of how to operate various function of the table, as well as educational testing modules are provided to aid in the demonstrations. The modular steel structure was designed to achieve a weak-column strong-beam failure mechanism while being able to be push well past its flexural yielding by the shaking table. Non-linear analysis was carried out using OpenSEES to characterize the behavior of the structure.

4. Future Scope

This project mainly focuses on one low-rise structure with weak columns and strong beams that was built for this project. Two other structures were designed and described but were not built. It would be useful to construct these structures. One is a low-rise (four story structure) with a strong-column weak-beam behavior while the other is a higher-rise structure with a strong-column weak-beam behavior. Other variations on the existing structure can be explored. Stiffeners can be installed in the structure to study the stiffened response and identify its advantage over the un-stiffened structure. Dampers can be installed within the structure and the damped response can also be monitored. Additionally, accelerometers and displacement transducers can be attached to the structure to verify simulation results.

5. Appendix

Modal Analysis Check

The structure is idealized as a lumped mass system and modal analysis is carried out to verify the results obtained from OpenSEES analysis. Mass is assumed to be concentrated at each floor and is taken equal to the additional weights attached to the structure. Beams are considered rigid for simplicity. Mass and stiffness matrices are determined using a shape function, which represents first mode of the structure.

$w := 1.65$

$I := 0.0000364$

$E := 29000000$

$\zeta n := 0.02$

$L := 12$

$ORIGIN := 1$

$$k := \begin{pmatrix} 2\cdot 12\dfrac{E\cdot I}{L^3} \\ 2\cdot 12\dfrac{E\cdot I}{L^3} \\ 2\cdot 12\dfrac{E\cdot I}{L^3} \\ 2\cdot 12\dfrac{E\cdot I}{L^3} \end{pmatrix}$$

$$k = \begin{pmatrix} 14.665 \\ 14.665 \\ 14.665 \\ 14.665 \end{pmatrix}$$

$$K := \begin{pmatrix} k_1 + k_2 & -k_2 & 0 & 0 \\ -k_2 & k_2 + k_3 & -k_3 & 0 \\ 0 & -k_3 & k_4 + k_3 & -k_4 \\ 0 & 0 & -k_4 & k_4 \end{pmatrix}$$

$$K = \begin{pmatrix} 29.33 & -14.665 & 0 & 0 \\ -14.665 & 29.33 & -14.665 & 0 \\ 0 & -14.665 & 29.33 & -14.665 \\ 0 & 0 & -14.665 & 14.665 \end{pmatrix}$$

$$\psi := \begin{pmatrix} 3.30 \\ 6.4 \\ 8.71 \\ 9.94 \end{pmatrix}$$

$$m := \begin{pmatrix} \frac{w}{386} & 0 & 0 & 0 \\ 0 & \frac{w}{386} & 0 & 0 \\ 0 & 0 & \frac{w}{386} & 0 \\ 0 & 0 & 0 & \frac{w}{386} \end{pmatrix}$$

$$m = \begin{pmatrix} 4.275 \times 10^{-3} & 0 & 0 & 0 \\ 0 & 4.275 \times 10^{-3} & 0 & 0 \\ 0 & 0 & 4.275 \times 10^{-3} & 0 \\ 0 & 0 & 0 & 4.275 \times 10^{-3} \end{pmatrix}$$

$$m1 := \psi^T \cdot m \cdot \psi$$

$$m1 = 0.968$$

$k1 := \psi^T \cdot K \cdot \psi$

$k1 = 401.077$

$$U := \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$L1 := \psi^T \cdot m \cdot U$

$L1 = 0.121$

$$\omega n := \sqrt{\frac{k1}{m1}}$$

$\omega n = 20.352$

$$Tn := \frac{2(\pi)}{\omega n} = 0.309$$

$A := 1.5 \cdot 386 = 579$

$$\Gamma := \frac{L1}{m1} = 0.125$$

$$f := \Gamma \cdot m \cdot \psi \cdot A = \begin{pmatrix} 1.022 \\ 1.982 \\ 2.698 \\ 3.079 \end{pmatrix}$$

$$h := \begin{pmatrix} 12 \\ 24 \\ 36 \\ 48 \end{pmatrix}$$

$Vb := L1 \cdot \Gamma \cdot A = 8.782$ lb

Note:

From the check it can be seen that the natural period is in close agreement with OpenSEES output. From modal analysis (beams idealized as rigid elements) we have, Tn = 0.31 s and from OpenSEES analysis we have, Tn = 0.33 s.

The value for base shear from modal analysis is 8.8 lb. For base shear values greater than 8.8 lb we should expect some yielding, as seen from Kobe and Northridge OpenSEES outputs. For both the ground motions, yielding is observed for a base shear value of 12 lb as indicated by the pushover analysis.

OpenSEES Code: Pushover Analysis

```
# SET UP ----------------------------------------------------------------------------

wipe;                                   # clear opensees model
model basic -ndm 2 -ndf 3;              # 2 dimensions, 3 dof per node
file mkdir 2Dnl1out;                    # create data directory

# define GEOMETRY --------------------------------------------------------------
set m  [expr 0.825/386.0]

set WallThk 0.0478 ;        #column section
set WallWidth 4
set SlabThk 0.1046 ;        #beam section
set SlabWidth 4
set AngleThk 0.125 ;        #connection angle
set AngleWidth 4
set BAngleThk 0.250 ;       #base angle
set BAngleWidth 4
set h 12;                   #story height
set L 8;                    #beam length
set x1 2;                    #rigid offset at base
set x 1;                    #rigid offset elsewhere

set lpw [expr 2*$WallThk];  #plastic hinge length for column
set lps [expr 2*$SlabThk];  #plastic hinge length for beam
set m [expr $h-2*$x-2*$lpw];        #remaining portion of the column after excluding the rigid offsets and hinge lengths
set n [expr $L-2*$x-2*$lps];        #remaining portion of the beam after excluding the rigid offsets and hinge lengths
set m1 [expr $h-$x-$x1-2*$lpw];     #remaining portion of the first floor column after excluding the rigid offsets and hinge lengths

# calculated geometric parameters
set Aw [expr $WallThk*$WallWidth]
set Izw [expr 1./12.*$WallWidth*pow($WallThk,3)]
set Zw [expr $Izw/($WallThk/2)]
set As [expr $SlabThk*$SlabWidth]
set Izs [expr 1./12.*$SlabWidth*pow($SlabThk,3)]
set Zs [expr $Izs/($SlabThk/2)]
set Aa [expr $AngleThk*$AngleWidth]
set Ia [expr 1./12.*$AngleWidth*pow($AngleThk,3)]
set Za [expr $Ia/($AngleThk/2)]
```

```
set Aba [expr $BAngleThk*$BAngleWidth]
set Iba [expr 1./12.*$BAngleWidth*pow($BAngleThk,3)]
set Zba [expr $Iba/($BAngleThk/2)]


set Fy 36000
set Es 29000000;                # Steel Young's Modulus
set nu 0.3;
set Gs [expr $Es/2./[expr 1+$nu]];  # Torsional stiffness Modulus
set b 0.01
set matID 1


set EACol [expr $Es*$Aw];                  # EA, for axial-force-strain relationship
set EABeam [expr $Es*$As];                 # EA, for axial-force-strain relationship
set EAAngle [expr $Es*$Aa]
set EABAngle [expr $Es*$Aba]
set EICol [expr $Es*$Izw];                 # EA, for moment-curvature relationship
set EIBeam [expr $Es*$Izs];                # EA, for moment-curvature relationship
set EIAngle [expr $Es*$Ia]
set EIBAngle [expr $Es*$Iba]
set MyCol [expr $Zw*$Fy]
set MyBeam [expr $Zs*$Fy]
set MyAngle [expr $Za*$Fy]
set MyBAngle [expr $Zba*$Fy]


# Define ELEMENTS & SECTIONS  -------------------------------------------------------------
set ColMatTagFlex 2
set BeamMatTagFlex 3
set BeamMatTagAxial 4;        # assign a tag number to the column flexural behavior
set ColMatTagAxial 5;         # assign a tag number to the column axial behavior
set AngleMatTagFlex 6
set BAngleMatTagFlex 7
set AngleMatTagAxial 8
set BAngleMatTagAxial 9
set ColSecTag 10;             # assign a tag number to the column section tag
set BeamSecTag 11;            # assign a tag number to the beam section tag
set AngleSecTag 12
set BAngleSecTag 13


#   separate columns and beams, in case of P-Delta analysis for columns
set IDColTransf 14; # all columns
```

```
set IDBeamTransf 15; # all beams
set IDAngleTransf 16

set IDBAngleTransf 17
set ColTransfType Linear ;                    # options, Linear PDelta Corotational
geomTransf $ColTransfType $IDColTransf ;       # only columns can have PDelta
effects (gravity effects)
geomTransf Linear $IDBeamTransf
geomTransf Linear $IDAngleTransf
geomTransf Linear $IDBAngleTransf

uniaxialMaterial Steel02  $ColMatTagFlex $MyCol $EICol $b
uniaxialMaterial Steel02  $BeamMatTagFlex $MyBeam $EIBeam $b
uniaxialMaterial Steel02  $AngleMatTagFlex $MyAngle $EIAngle $b
uniaxialMaterial Steel02  $BAngleMatTagFlex $MyBAngle $EIBAngle $b

uniaxialMaterial Elastic $ColMatTagAxial $EACol;        # this is not used as a
material, this is an axial-force-strain response
section Aggregator $ColSecTag $ColMatTagAxial P $ColMatTagFlex Mz;
# combine axial and flexural behavior into one section (no P-M interaction here)
uniaxialMaterial Elastic $BeamMatTagAxial $EABeam;
# this is not used as a material, this is an axial-force-strain response

section Aggregator $BeamSecTag $BeamMatTagAxial P $BeamMatTagFlex Mz;
# combine axial and flexural behavior into one section (no P-M interaction here)
uniaxialMaterial Elastic $AngleMatTagAxial $EAAngle;
# this is not used as a material, this is an axial-force-strain response

section Aggregator $AngleSecTag $AngleMatTagAxial P $AngleMatTagFlex Mz;
# combine axial and flexural behavior into one section (no P-M interaction here)

uniaxialMaterial Elastic $BAngleMatTagAxial $EABAngle;
# this is not used as a material, this is an axial-force-strain response
section Aggregator $BAngleSecTag $BAngleMatTagAxial P $BAngleMatTagFlex Mz;
# combine axial and flexural behavior into one section (no P-M interaction here)


# nodal coordinates:
node 11 0 0;                                  # node#, X Y
node 12 0 $x1
node 13 0 [expr $x1+$lpw]
node 14 0 [expr $x1+$lpw+$m1]
node 15 0 [expr $x1+$lpw+$m1+$lpw]
node 16 $L 0
```

```
node 17 $L $x1
node 18 $L [expr $x1+$lpw]
node 19 $L [expr $x1+$lpw+$m1]
node 20 $L [expr $x1+$lpw+$m1+$lpw]
node 101 0 $h
node 102 $x $h
node 103 [expr $x+$lps] $h
node 104 [expr $x+$lps+$n] $h
node 105 [expr $x+$lps+$n+$lps] $h
node 106 $L $h


node 21 0 [expr $h+$x]
node 22 0 [expr $h+$x+$lpw]
node 23 0 [expr $h+$x+$lpw+$m]
node 24 0 [expr $h+$x+$lpw+$m+$lpw]
node 25 $L [expr $h+$x]
node 26 $L [expr $h+$x+$lpw]
node 27 $L [expr $h+$x+$lpw+$m]
node 28 $L [expr $h+$x+$lpw+$m+$lpw]


node 201 0 [expr 2*$h]
node 202 $x [expr 2*$h]
node 203 [expr $x+$lps] [expr 2*$h]
node 204 [expr $x+$lps+$n] [expr 2*$h]
node 205 [expr $x+$lps+$n+$lps] [expr 2*$h]
node 206 $L [expr 2*$h]


node 31 0 [expr 2*$h+$x]
node 32 0 [expr 2*$h+$x+$lpw]
node 33 0 [expr 2*$h+$x+$lpw+$m]
node 34 0 [expr 2*$h+$x+$lpw+$m+$lpw]
node 35 $L [expr 2*$h+$x]
node 36 $L [expr 2*$h+$x+$lpw]
node 37 $L [expr 2*$h+$x+$lpw+$m]
node 38 $L [expr 2*$h+$x+$lpw+$m+$lpw]


node 301 0 [expr 3*$h]
node 302 $x [expr 3*$h]
node 303 [expr $x+$lps] [expr 3*$h]
node 304 [expr $x+$lps+$n] [expr 3*$h]
```

```
node 305 [expr $x+$lps+$n+$lps] [expr 3*$h]
node 306 $L [expr 3*$h]



node 41 0 [expr 3*$h+$x]
node 42 0 [expr 3*$h+$x+$lpw]
node 43 0 [expr 3*$h+$x+$lpw+$m]
node 44 0 [expr 3*$h+$x+$lpw+$m+$lpw]
node 45 $L [expr 3*$h+$x]
node 46 $L [expr 3*$h+$x+$lpw]
node 47 $L [expr 3*$h+$x+$lpw+$m]
node 48 $L [expr 3*$h+$x+$lpw+$m+$lpw]


node 401 0 [expr 4*$h]
node 402 $x [expr 4*$h]
node 403 [expr $x+$lps] [expr 4*$h]
node 404 [expr $x+$lps+$n] [expr 4*$h]
node 405 [expr $x+$lps+$n+$lps] [expr 4*$h]
node 406 $L [expr 4*$h]


# Single point constraints -- Boundary Conditions
fix 11 1 1 1;                  # node DX DY RZ
fix 16 1 1 1;                  # node DX DY RZ


# nodal masses:
mass 101 $m $m $m;             # node Mx My Mz, Mass=Weight/g.
mass 106 $m $m $m
mass 201 $m $m $m
mass 206 $m $m $m
mass 301 $m $m $m
mass 306 $m $m $m
mass 401 $m $m $m
mass 406 $m $m $m


# Define ELEMENTS ----------------------------------------------------------

set numIntgrPts 2;             # number of integration points for end offsets and plastic
hinge lengths
```

set numIntgrPts1 5;   # number of integration points for force-based element

connectivity:
the elements are numbered story-wise, starting with the right column (5 elements), then the left column (5 elements) and then the story beam (5 elements).
Thus, every story comprises of 15 elements... 1-15, 16-30, 31-45, 45-60

```
element nonlinearBeamColumn 1 11 12 $numIntgrPts $BAngleSecTag $IDBAngleTransf ;    # self-explanatory when using variables
element nonlinearBeamColumn 2 12 13 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 3 13 14 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 4 14 15 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 5 15 101 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 6 16 17 $numIntgrPts $BAngleSecTag $IDBAngleTransf
element nonlinearBeamColumn 7 17 18 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 8 18 19 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 9 19 20 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 10 20 106 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 11 101 102 $numIntgrPts $AngleSecTag $IDColTransf
element nonlinearBeamColumn 12 102 103 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 13 103 104 $numIntgrPts1 $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 14 104 105 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 15 105 106 $numIntgrPts $AngleSecTag $IDAngleTransf


element nonlinearBeamColumn 16 101 21 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 17 21 22 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 18 22 23 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 19 23 24 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 20 24 201 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 21 106 25 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 22 25 26 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 23 26 27 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 24 27 28 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 25 28 206 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 26 201 202 $numIntgrPts $AngleSecTag $IDAngleTransf
```

```
element nonlinearBeamColumn 27 202 203 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 28 203 204 $numIntgrPts1 $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 29 204 205 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 30 205 206 $numIntgrPts $AngleSecTag $IDAngleTransf


element nonlinearBeamColumn 31 201 31 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 32 31 32 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 33 32 33 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 34 33 34 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 35 34 301 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 36 206 35 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 37 35 36 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 38 36 37 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 39 37 38 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 40 38 306 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 41 301 302 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 42 302 303 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 43 303 304 $numIntgrPts1 $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 44 304 305 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 45 305 306 $numIntgrPts $AngleSecTag $IDAngleTransf


element nonlinearBeamColumn 46 301 41 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 47 41 42 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 48 42 43 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 49 43 44 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 50 44 401 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 51 306 45 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 52 45 46 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 53 46 47 $numIntgrPts1 $ColSecTag $IDColTransf
element nonlinearBeamColumn 54 47 48 $numIntgrPts $ColSecTag $IDColTransf
element nonlinearBeamColumn 55 48 406 $numIntgrPts $AngleSecTag $IDAngleTransf
```

element nonlinearBeamColumn 56 401 402 $numIntgrPts $AngleSecTag $IDAngleTransf
element nonlinearBeamColumn 57 402 403 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 58 403 404 $numIntgrPts1 $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 59 404 405 $numIntgrPts $BeamSecTag $IDBeamTransf
element nonlinearBeamColumn 60 405 406 $numIntgrPts $AngleSecTag $IDAngleTransf

Define RECORDERS ---
recorder Node -file 2Dnl1out/DFree.out -time -node 101 106 201 206 301 306 401 406 -dof 1 2 3 disp;   # displacements of free nodes
recorder Node -file 2Dnl1out/DFree406.out -time -node 406 -dof 1 disp
recorder Node -file 2Dnl1out/DBase.out -time -node 11 12 -dof 1 2 3 disp;
displacements of support nodes
recorder Node -file 2Dnl1out/RBaseX1.out -time -node 11 -dof 1 reaction;
support reaction
recorder Node -file 2Dnl1out/RBaseX2.out -time -node 12 -dof 1 reaction
recorder Drift -file 2Dnl1out/Drift.out -time -iNode 11 -jNode 401 -dof 1 -perpDirn 2 ;
lateral drift

# define GRAVITY --
pattern Plain 1 Linear {
eleLoad -ele 11 12 13 14 15 26 27 28 29 30 41 42 43 45 56 57 58 59 60 -type -beamUniform -1.17e-1 ; # distributed superstructure-weight on beam
}

# define LATERAL load --
# Lateral load pattern
pattern Plain 2 Linear {
load 101 2.0 0.0 0.0;   # node FX FY MZ -- representative lateral load at top nodes
load 201 4.0 0.0 0.0;   # place 1/2 of the weight for each node to get shear coefficient
load 301 6.0 0.0 0.0
load 401 8.0 0.0 0.0
}

STATIC PUSHOVER ANALYSIS ---

we need to set up parameters that are particular to the model.
set IDctrlNode 406;		# node where displacement is read for displacement control
set IDctrlDOF 1;	# degree of freedom of displacement read for displacement control
set Dmax 20;		# maximum displacement of pushover. push to 10% drift.
set Dincr 0.1;		# displacement increment for pushover. you want this to be very small, but not too small to slow down the

----------- set up analysis parameters
CONSTRAINTS handler -- Determines how the constraint equations are enforced in the analysis (http://opensees.berkeley.edu/OpenSEES/manuals/usermanual/617.htm)
#Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#Transformation Method -- Performs a condensation of constrained degrees of freedom

constraints Plain;
#DOF NUMBERER (number the degrees of freedom in the domain): (http://opensees.berkeley.edu/OpenSEES/manuals/usermanual/366.htm)
determines the mapping between equation numbers and degrees-of-freedom
Plain -- Uses the numbering provided by the user
RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
numberer Plain
SYSTEM (http://opensees.berkeley.edu/OpenSEES/manuals/usermanual/371.htm)
Linear Equation Solvers (how to store and solve the system of equations in the analysis)
-- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#	ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#	BandGeneral -- Direct solver for banded unsymmetric matrices
#	BandSPD -- Direct solver for banded symmetric positive definite matrices
#	SparseGeneral -- Direct solver for unsymmetric sparse matrices
#	SparseSPD -- Direct solver for symmetric sparse matrices
#	UmfPack -- Direct UmfPack solver for unsymmetric matrices
system BandGeneral

TEST: # convergence test to

Convergence TEST (http://opensees.berkeley.edu/OpenSEES/manuals/usermanual/360.htm)
-- Accept the current state of the domain as being on the converged solution path
-- determine if convergence has been achieved at the end of an iteration step
NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration

set Tol 1.e-8; # Convergence Test: tolerance
set maxNumIter 6; # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
set printFlag 0; # Convergence Test: flag used to print information on convergence (optional) # 1: print information on each step;
set TestType EnergyIncr ; # Convergence-test type
test $TestType $Tol $maxNumIter $printFlag;

Solution ALGORITHM: -- Iterate from the last time step to the current (http://opensees.berkeley.edu/OpenSEES/manuals/usermanual/682.htm)
Linear -- Uses the solution at the first iteration and continues
Newton -- Uses the tangent at the current iteration to iterate to convergence
ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence

set algorithmType Newton
algorithm $algorithmType;

Static INTEGRATOR: -- determine the next time step for an analysis (http://opensees.berkeley.edu/OpenSEES/manuals/usermanual/689.htm)
LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
Arc Length -- Specifies the incremental arc-length of the load-displacement path
Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
Newmark -- The two parameter time-stepping method developed by Newmark
HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement

integrator DisplacementControl $IDctrlNode $IDctrlDOF $Dincr

ANALYSIS -- defines what type of analysis is to be performed (http://opensees.berkeley.edu/OpenSEES/manuals/usermanual/324.htm)
Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.

```
analysis Static
#  ---------------------------------    perform Static Pushover Analysis

set Nsteps [expr int($Dmax/$Dincr)];       # number of pushover analysis steps
set ok [analyze $Nsteps];                 # this will return zero if no convergence problems were encountered

# ---------------------------------- in case of convergence problems

if {$ok != 0} {
# change some analysis parameters to achieve convergence
# performance is slower inside this loop

	set ok 0;
	set controlDisp 0.0;		# start from zero

	set D0 0.0;			# start from zero
	set Dstep [expr ($controlDisp-$D0)/($Dmax-$D0)]
	while {$Dstep < 1.0 && $ok == 0} {

		set controlDisp [nodeDisp $IDctrlNode $IDctrlDOF ]
		set Dstep [expr ($controlDisp-$D0)/($Dmax-$D0)]
		set ok [analyze 1 ]

		if {$ok != 0} {

			puts "Trying Newton with Initial Tangent .."
			test NormDispIncr   $Tol 2000  0
			algorithm Newton -initial
			set ok [analyze 1 ]
			test $TestType $Tol $maxNumIter  0
```

```
            algorithm $algorithmType
        }

        if {$ok != 0} {

            puts "Trying Broyden .."
            algorithm Broyden 8
            set ok [analyze 1 ]
            algorithm $algorithmType
        }

        if {$ok != 0} {

            puts "Trying NewtonWithLineSearch .."
            algorithm NewtonLineSearch .8
            set ok [analyze 1 ]
            algorithm $algorithmType
        }
    }
    };    # end if ok !0


puts "DonePushover"
```

OpenSEES Code: Eigenvalue Analysis

The element formulation is same as shown in the pushover analysis. The code that follows will change as shown below:

```
# record eigenvectors----------------------
for { set k 1 } { $k <= $numModes } { incr k } {
recorder Node -file [format "modes/mode%i.out" $k] -nodeRange 1 10 -dof 1 2 3 "eigen
$k"
}

# perform eigen analysis-----------------------------
set lambda [eigen  $numModes];

# calculate frequencies and periods of the structure---------------------------------------
set omega {}
set f {}
set T {}
set pi 3.141593
```

```
foreach lam $lambda {

	lappend omega [expr sqrt($lam)]
	lappend f [expr sqrt($lam)/(2*$pi)]
	lappend T [expr (2*$pi)/sqrt($lam)]
}

puts "periods are $T"
# write the output file cosisting of periods--------------------------------------------
set period "modes/Periods.txt"
set Periods [open $period "w"]
foreach t $T {
	puts $Periods " $t"
}
close $Periods

# Run a one step gravity load with no loading (to record eigenvectors)-----------------------
-----------------------------------------------
integrator LoadControl 0 1 0 0
# Convergence test
# tolerance maxIter displayCode
test EnergyIncr		1.0e-10	100	0

# Solution algorithm
algorithm Newton

# DOF numberer
numberer RCM

# Constraint handler
constraints Transformation

# System of equations solver
system ProfileSPD

analysis Static
set res [analyze 1]

if {$res < 0} {
   puts "Modal analysis failed"
}
```

OpenSEES Code: Ground Motions

The element formulation is same as shown in the pushover analysis. The code that follows will change as shown below:

```
# define GRAVITY -------------------------------------------------------------
pattern Plain 1 Linear {
eleLoad -ele 3 6 9 12 -type -beamUniform -1.17e-1 ; # distributed superstructure-weight
on beam
}

constraints Plain;                          # how it handles boundary conditions
numberer Plain;      # renumber dof's to minimize band-width (optimization), if you
want to
system BandGeneral; # how to store and solve the system of equations in the analysis

# Uniform Earthquake ground motion (uniform acceleration input at all support nodes)
set GMdirection 1;                          # ground-motion direction
set GMfile "Nridge" ;                 # ground-motion filenames
set GMfact 1;                         # ground-motion scaling factor
source LibUnits.tcl

# set up ground-motion-analysis parameters
set DtAnalysis [expr 0.01*$sec];      # time-step Dt for lateral analysis
set TmaxAnalysis     [expr 10.*$sec];       # maximum duration of ground-motion
analysis -- should be 50*$sec

# ----------- set up analysis parameters
source LibAnalysisDynamicParameters.tcl;
#constraintsHandler, DOFnumberer,system ofequations, convergenceTest,
solutionAlgorithm, integrator

# define DAMPING-----------------------------------------------------------------------
# apply Rayleigh DAMPING from $xDamp
# D=$alphaM*M + $betaKcurr*Kcurrent + $betaKcomm*KlastCommit +
$beatKinit*$Kinitial

set xDamp 0.02;                             # 2% damping ratio
set lambda [eigen 1];                 # eigenvalue mode 1
set omega [expr pow($lambda,0.5)];
set alphaM 0.;                        # M-prop. damping; D = alphaM*M
set betaKcurr 0.;      # K-proportional damping;    +beatKcurr*KCurrent
```

set betaKcomm [expr 2.*$xDamp/($omega)]; # K-prop. damping parameter; +betaKcomm*KlastCommitt
set betaKinit 0.; # initial-stiffness proportional damping +beatKinit*Kini
rayleigh $alphaM $betaKcurr $betaKinit $betaKcomm;
RAYLEIGH damping

-------------------------------- perform Dynamic Ground-Motion Analysis
the following commands are unique to the Uniform Earthquake excitation

set IDloadTag 400; # for uniformSupport excitation

read a PEER strong motion database file, extracts dt from the header and converts the file
to the format OpenSEES expects for Uniform/multiple-support ground motions

source ReadSMDFile.tcl; # read in procedure Multinition

Uniform EXCITATION: acceleration input
set inFile $GMfile.at2
set outFile $GMfile.g3; # set variable holding new filename (PEER files have .at2/dt2 extension)
ReadSMDFile $inFile $outFile dt; # call procedure to convert the ground-motion file
set GMfatt [expr $g*$GMfact]; # data in input file is in g Unifts -- ACCELERATION TH
set AccelSeries "Series -dt $dt -filePath $outFile -factor $GMfatt";# time series information
pattern UniformExcitation $IDloadTag $GMdirection -accel $AccelSeries ;
 # create Unifform excitation
set Nsteps [expr int($TmaxAnalysis/$DtAnalysis)];
set ok [analyze $Nsteps $DtAnalysis]; # actually perform analysis; returns ok=0 if analysis was successful

if {$ok != 0} { ; # analysis was not successful.

---
change some analysis parameters to achieve convergence
performance is slower inside this loop
Time-controlled analysis
 set ok 0;
 set controlTime [getTime];
 while {$controlTime < $TmaxAnalysis && $ok == 0} {
 set controlTime [getTime]
 set ok [analyze 1 $DtAnalysis]

```
        if {$ok != 0} {

            puts "Trying Newton with Initial Tangent .."
            test NormDispIncr   $Tol 1000  0
            algorithm Newton -initial
            set ok [analyze 1 $DtAnalysis]
            test $testTypeDynamic $TolDynamic $maxNumIterDynamic  0
            algorithm $algorithmTypeDynamic
        }

        if {$ok != 0} {
            puts "Trying Broyden .."
            algorithm Broyden 8
            set ok [analyze 1 $DtAnalysis]
            algorithm $algorithmTypeDynamic
        }

        if {$ok != 0} {
            puts "Trying NewtonWithLineSearch .."
            algorithm NewtonLineSearch .8
            set ok [analyze 1 $DtAnalysis]
            algorithm $algorithmTypeDynamic
        }
    }
};     # end if ok !0


puts "Ground Motion Done. End Time: [getTime]"
```

6. References

1. http://www.quanser.com/english/html/earthquake/Shake Table II .
2. Quanser (2008) “Specialty Plant: Shake Table II Position Control and Earthquake Analysis”.
3. MATLAB and Simulink : MathWorks Release 2008b
4. McKenna, F., Fenves, G. L., Scott, M. H., and Jeremie, B. (2000). "Open System for Earthquake Engineering Simulation, OpenSEES." Berkeley, California.
5. Menegotto, M., and Pinto, P. E. "Method of Analysis for Cyclically Loaded Reinforced Concrete Plane Frames Including Changes in Geometry and Nonelastic Behavior of Elements under Combined Normal Force and Bending." Proceedings of IABSE Symposium on Resistance and Ultimate Deformability of Structures Acted on by Well-Defined Repeated Loads, Lisbon.
6. Spacone, E., Filippou, F. C., and Taucer, F. F. (1996). "Fibre Beam-Column Model for Non-Linear Analysis of R/C Frames: Part I. Formulation." *Earthquake Engineering & Structural Dynamics*, 25(7), 711-725.